# database and text editor

## SUPERBASE AND AMIGA WORKBENCH

To run Superbase under Amiga Workbench, you need to license and install Amiga Workbench Release 1.3 or later.

## LIMITED WARRANTY

If there is a physical defect in the Superbase program disk or manual, Precision Software Limited will replace the defective item free of charge, provided you return the item to be replaced with proof of purchase to Precision Software Limited within 90 days of the date of purchase. In some countries the replacement period may be different; check with your software supplier.

The contents of this manual are subject to change without notice and do not represent any commitment on the part of Precision Software Limited.

No warranty is made with respect to Superbase, its quality or performance. Any and all warranties for merchantability and/or fitness for a particular purpose are expressly excluded. The Superbase software is sold 'as is,' and you, the purchaser, are assuming the entire risk as to its quality and performance.

In no event shall Precision Software Limited be liable or responsible to the purchaser or any other person or entity for direct, indirect, special, incidental, or consequential damages, including but not limited to interruption of service and loss of business or anticipatory profits, resulting from any defect in or misunderstanding of the software or its documentation, even if Precision Software Limited has been advised of the possibility of such damages.

The warranty and remedies set out above are exclusive and in lieu of all others, oral or written, express or implied.

Some countries do not allow the exclusion or limitation of implied warranties or liabilities for incidental or consequential damages, so the above limitation or exclusion may not apply to you.

## REGISTRATION AND SUPPORT

Please complete and return the Registration Card, which is included in your Superbase package.

*Only registered users are eligible for 30 days' free software support from time of registration. If you do not return the card, we will not be able to help you.*

Please remember, when requesting support, to make available:

- Your registration number. See your program disk or registration card.
- The version number of your copy of Superbase. This is displayed in the 'About' requester available from the Project menu.
- A clear statement of your problem.

While we are proud of our ability to help users solve their problems, we ask you to remember that only a precise description of a problem can lead to a precise solution.

# Before You Begin ...

## Read Me First

With this package you will find a set of pages titled IMPORTANT: READ ME FIRST. You should read them before starting to use Superbase. The Read Me First pages include:

- Package Contents check list.
- Instructions for installing Superbase.
- Instructions for running Superbase.

You should have already installed Superbase. If you haven't done so, the procedure is simply a matter of running the Setup program on the Superbase setup disk:

1. Insert the Superbase Setup disk in any floppy drive.

2. Open the disk window by double-clicking on the disk icon.

3. Double-click on the "Setup" icon.

4. Follow the installation instructions as they appear on screen.

## README.TXT

The Superbase program disk includes a document called README.TXT. This document describes the changes made to Superbase between the time your Superbase User Guide went to press and the time the final modifications were made to the software. README.TXT is an ASCII text file. You can read it either with the Superbase Text Editor or by double-clicking on its icon.

## The Superbase Documentation

The Superbase manuals are designed to allow several ways of becoming familiar with the program.

Volume 1 is a User Guide for the Database and Text Editor, with an introductory tutorial and a reference summary. The User Guide is structured around practical

topics, so that you can locate 'how to' instructions easily. The Table of Contents shows the arrangement of topics.

Volume 2 contains a User Guide for the Form Designer, a Keyword Reference Guide for Superbase's Database Management Language (DML), and Advanced Topics chapters.

Volume 2 also includes an Applications Guide, in which a range of worked examples shows how Superbase can be used to build effective database applications.

# Compatibility

## Superbase Professional 3.02 and Earlier

Superbase Professional 4 Amiga represents the latest in Superbase software technology, and is functionally equivalent in most respects to Superbase 4 Windows. Because it is not derived from Superbase Professional 3, some existing applications developed under that system may require some minor modifications to make them work with Superbase Professional 4 Amiga.

Database files, text files, forms, and other ancillary files created under Superbase Professional 3 should require no modifications. However, DML programs may require some changes. See the README.TXT file on the program disk for details.

Files, forms and programs created under Superbase Professional 4 Amiga should be regarded as *not* compatible with Superbase Professional 3. However, if you use a file originally created in Superbase Professional 3 in Superbase Professional 4 Amiga you may continue to read it with Superbase Professional 3 provided you do not reorganize it. Files used in this way may not take advantage of any of the features introduced with Superbase Professional 4 Amiga.

## Superbase 4 Windows

Database files, text files, and other ancillary files created under Superbase 4 Windows 1.2 or 1.3 may be used with Superbase Professional 4 Amiga without modification.

DML programs may require some changes, as some commands are irrelevant in the Amiga environment. DML programs written under Superbase Professional 4 Amiga will be largely compatible with the Windows version.

However, forms created in either environment are not compatible with the other.

See the README.TXT file on the program disk for further details.

# CONTENTS

# 1 GETTING TO KNOW SUPERBASE 4

## The Workbench Database

Superbase is a comprehensive, general purpose relational database management system designed to operate in the Commodore Amiga Workbench graphical environment.

Its intuitive interface consistently employs the standard Amiga Workbench elements: icons, tools, gadgets, windows, requesters, directories (or drawers), projects and so on. This simplifies the learning process both for users already familiar with Workbench and for those using a Workbench application for the first time.

In common with other Workbench applications, Superbase is flexible to configure and easy to install. It can be driven either using a mouse or from the keyboard. It benefits from the wide and well supported library of Workbench display and output device drivers. In addition, its bit-mapped orientation makes it possible for Superbase to extend the range of functions offered by character-oriented databases in valuable new directions.

Firstly, it can treat many types of graphical objects as data, including scanned or drawn images and drawings, thus dramatically increasing the information 'bandwidth.' Secondly it adopts a what-you-see-is-what-you-get approach to the design of business forms and reports. Graphical design elements may be incorporated into forms for more meaningful and motivating data entry and enquiry screens as well as published quality printed output.

As a Workbench application, Superbase runs alongside other Workbench graphical and business applications, and can exchange data with them.

### The Multi-Layered Database

Superbase is a multi-layered database system, incorporating both user-oriented layers for the ready manipulation of straightforward filing systems and developer-oriented layers for the design and programming of custom application suites based on sophisticated relational file structures. This multi-layered approach provides the user with a number of possible techniques for achieving any particular result.

For the non-technical user, Superbase provides an intuitive primary work surface centered on the familiar metaphor of VCR controls. These controls provide immediate browsing access to data records in any sequence – get next, fast forward, get last, rewind, and so on – in a manner similar to the controls of a video cassette recorder.

At an intermediate level, Superbase offers powerful relational capabilities without programming, including cross-file lookups and calculations as part of the file definition process, and the formulation of re-usable multi-file queries.

A separate easy-to-use forms design surface provides extensive tools for creating WYSIWYG forms and reports from a wide range of design objects, including lines, boxes, areas, text, digitized images and other graphics, fields from any file, calculation formulas, and command calls. Forms can incorporate one-to-many relationships, with relations being displayed graphically as part of the forms design process. Forms can then be accessed for browsing and data entry from within the primary work surface.

Superbase offers the technical user the further layer of a full-scale Database Management Language (DML) as well as re-usable macros and function key assignments. Many complex activities, such as one-to-many forms browsing or multi-column label printing, may be accomplished either from the menu, or by typing a single-line command, or by assigning it to a function key, or by incorporating the command in a DML program.

Superbase accordingly offers the developer a number of distinct but complementary approaches to application development, including a non-procedural approach through the forms design surface and a procedural approach through the Database Management Language. These approaches are complementary, as forms can both invoke and be invoked from programmed procedures.

The flexibility of the multi-layered approach appeals equally to the busy executive looking for a powerful tool capable of producing immediate results and to the programmer seeking to develop a finely-tuned custom system within an easy-to-use graphical environment.

All the browsing, relational reporting and forms design tools provided with Superbase operate directly on files in dBase II and dBase III format, as well as files created or imported using Superbase.

The first volume of this manual deals with the set of menu-driven facilities which together constitute a definitive productivity tool for data management. The development-oriented tools and facilities are covered in the second volume.

# The Primary Work Surface

Superbase presents the user with an everyday work surface which combines instant familiarity with near infinite flexibility. You can define and refine file structures, even when a file contains data, without any need to rebuild them. There are no practical limits to the numbers of fields or to the record length.

You can retrieve data in any form or order you wish. Using simple menu commands, you select the required fields, place them in any order, choose page or table orientation, and switch to your preferred index sequence. The VCR-like controls at the foot of the work area allow you to scan files, look up records, apply filter conditions, or simply step through your records one at a time. Information retrieved in this way can be printed with a single menu selection.

Included as part of the primary work surface is the Superbase Text Editor. This handles files in ASCII or ANSI format, as well as its own format. The Text Editor is intended as a no-frills tool for supporting database operations, and includes cut and paste, search, and a direct merge interface to data files.

## The Secondary Database Management Surface

Superbase also offers a deeper layer of functionality, to which users naturally turn as they become more familiar with the primary work surface. The secondary database management surface allows users to exchange data with other programs via a range of import and export formats, design re- usable multi-file queries and global updates, and produce formatted reports. Using deep-structure programmable functions you can create linked databases with extensive data validation and sophisticated automatic calculations built in. A communications function gives you access to remote systems via modem.

## The Form Designer

The Form Designer module allows you to design forms for screen input, browsing, and printing, using a full range of graphic objects and editing tools. You may include fields from many files on a form, which may in turn have multiple pages. A block of fields may be defined as 'transaction lines,' so that one-to-many file relationships can be managed without programming. Field data may be displayed and printed in different fonts. Calculation formulas and commands may be incorporated to give the form powerful processing capabilities.

The Form Designer also includes a Report Generator, which allows you to build precise report definitions with simple point-and-click techniques. The report form creates a simple Superbase DML program, which is available to run immediately or for fine-tuning in the built-in Program Editor.

## System Automation and Customizing

When you are ready to automate your system, there are many techniques available to suit individual preferences. Global queries, updates, and label formats are all re-usable disk files. And the user-definable function key feature lets you create and store any number of sets of up to 40 function key strings, combining powerful DML syntax and functions into multi-statement command lines.

Beyond this layer of 're-usability,' Superbase offers, through its extensive programming language, DML, scope for the development of complete systems. Over 230 commands ranging from financial functions through multi-dimensional arrays to a multi-line query SELECT statement may be drawn on to create effective and comprehensive applications. User-definable pull-down menus may be configured for any application, and invoked from an auto-executing start-up sequence. Requesters with lists for point-and-click selection can be created with a single command. Your system can disable user interrupts, and there are full error-handling facilities. You may exchange data with other Workbench applications.

Superbase includes an integrated graphics file handling capability unique in its class. Bit-image PCX and GIF files, and ILBM, Dynamic HiRes and Dynamic HAM types of IFF files, can all be displayed from within the primary work surface, by defining a field as 'external' and storing filenames in it. External text can also be displayed. Using forms, data, images and text may be displayed and printed directly. This feature enables users to apply the power of database organization to the image files generated by scanners and digitizers as well as graphics software.

The rest of this chapter provides an introduction to the program, designed to get you working productively as quickly as possible. The other chapters in this volume of the manual deal with the tasks you are most likely to want to perform with Superbase.

## Loading Superbase

The procedure for loading Superbase is:

1.  Insert a Workbench disk in drive 0 and start up the Amiga.

2.  When the FastHD disc icon is displayed double-click on it.

3.  When the available tools are displayed double-click on the Superbase icon.

4.  When you are given a choice between Superbase (SBPRO4) and its Forms Designer (SBFD4) double-click on the Superbase icon.

## The Superbase Window

Superbase functions like most other Workbench applications. If you are already familiar with how such applications work, you will be able to skip over some of the following sections.

The **title bar** at the top of the window displays the name of the current Superbase File and Index. The pathname is also shown if there is one.

# The Superbase Menus

| Project | | Edit | | Record | |
|---|---|---|---|---|---|
| New >> | | Current | A E | New | A N |
| Open >> | | --------- | | Save | A S |
| Close >> | | Cut | A X | Duplicate | A D |
| Modify >> | | Copy | A C | Batch | A B |
| Save | | Paste | A V | --------- | |
| Remove >> | | | | Next external | A > |
| Print >> | | | | Previous external | A < |
| About.. | | | | | |
| Quit | A Q | | | | |

| Process | | Set | |
|---|---|---|---|
| Query >> | | Table view | A T |
| Update >> | | Page view | A P |
| Remove... | | Record view | A R |
| --------- | | Form | A F |
| Import... | | --------- | |
| Export... | | Field selection.. | A = |
| Split.. | | Paging | |
| Mail merge.. | | Show field names | |
| Labels >> | | --------- | |
| Reorganize | | Sytem options... | |
| | | Printer setup... | |
| | | Number format... | |
| | | Date format... | |
| | | Function keys >> | |

| Utilities | | DML | |
|---|---|---|---|
| Directory List | A L | Command.. | A A |
| Text Editor | A H | Run | A G |
| Status >> | | New | |
| Screen dump | | Open... | |
| Communications... | | Close | |
| --------- | | Edit | A / |
| Type... | | Save.. | |
| Delete... | | Save as... | |
| Rename... | | Remove... | |
| Copy... | | | |

### Ghost Items

Menu items that are temporarily unavailable are shown in gray script, and are known as 'ghost' items. We use this feature in Superbase, most obviously during this start up phase when no file is open.

To see this, press the right-hand mouse button and move the pointer to the Edit Menu. You'll see that all the items on this menu appear to be faint, as if only half the dots that make up each letter were being used. If you try to select a ghost item, nothing will happen. A quick look at the browsing controls at the bottom of the screen will reveal that most of the control buttons are also ghost items while no file is open.

### Selecting Commands

The way you select commands in Superbase is exactly the same as using the Workbench itself. You use the mouse as follows:

1.  Press the right hand mouse button to display the names of the available menus in the menu bar.

2.  Keep the button held down while you move the pointer to the menu item you want.

3.  Release the mouse button to run the selected command.

### Sub-menus

Many menu items are followed by a '»' symbol. Selecting any menu item so marked causes a sub-menu to be displayed in addition to the main menu. This happens whenever you need to make a further choice. For example, if you display the 'Project' menu and select 'Open' then a sub-menu is displayed allowing you to specify what you want to open: File, dBase File, Index or Form.

Keeping the mouse button held down, move the pointer into the sub-menu, point to the item you want, and release the button.

### Shortcuts

For some operations, Superbase provides shortcut key combinations, which you operate by holding down the right-hand Amiga key while pressing another key. These shortcuts allow you to select a menu command without having to pull down the menu first. So, for instance, you can display a list of file names by holding down the right-hand Amiga key while pressing the L key. Combination key shortcuts are shown on the menu to the right of the command name. Note that shortcuts will not work while a menu is on screen. (In some cases you may use either the left-hand or the right-hand Amiga key.)

# The Superbase Work Area

The work area is Superbase's permanent window onto your database. It's like a large worksheet, 273 columns across; the number of lines you can see depends on the size of your window. Superbase uses the work area to show data, either with or without a form. With a form, data from several files can be shown at once. Without a form, Superbase shows just the records in the current file, as well as the results of some other operations.
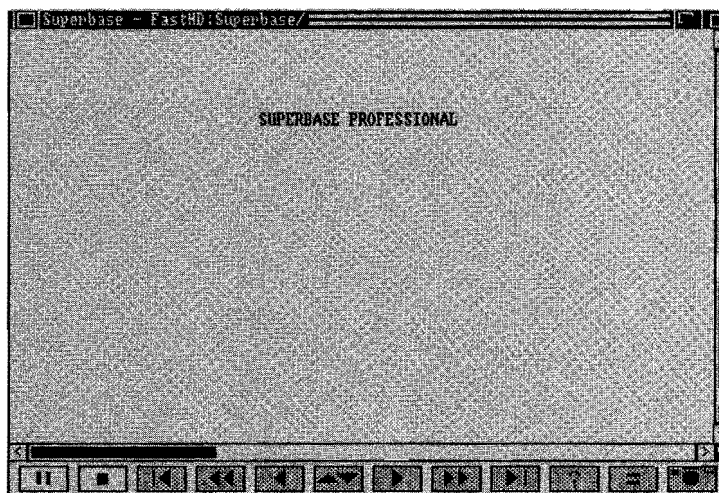


When a form is not being used, records may be shown in one of three default Views: Record, Page, and Table.

- **Record View** displays data one record at a time, with one field per line.
- **Table View** displays data one record per line, with up to 273 characters of data from each record.
- **Page View** displays data one record at a time in a page orientation, with fields dragged into position by the user.

The basic facilities for managing data allow you to tailor the way data is presented in the work area.

- You may **select fields** to view, and the order in which they appear.
- You may select any **index** to determine the sequence in which the records appear.
- You may select a subset of records from the file by means of a **filter**.

You use the **scroll bars** to move around the work area and look at data which is not immediately visible.

At the bottom of the window are Superbase's browsing controls, which give you immediate access to your database via a simple and familiar model. Most

of the twelve browsing controls are modeled on the controls for a video cassette or tape recorder, so you will find their meanings easy to learn and remember.

# Opening a File

In this section you will open a file, learn about Superbase requesters, practice switching Views, and use the browsing controls to step through records.

To begin, select Open File from the Project menu. Superbase displays the following requester box:
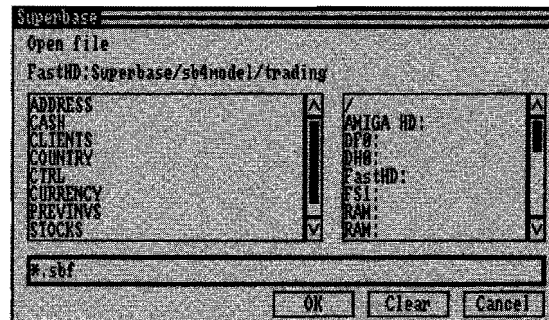


## Superbase Requester Boxes

Many of the commands on the menus are followed by '...'. These three dots indicate that after choosing the command, you will be presented with a requester box. We refer to these simply as requesters.

Superbase uses requesters to communicate information to you. For example, the requester you've just selected is the Open File requester.

Requesters like this one are used in most Superbase operations. They communicate messages to you and require you to perform a specific action. This action involves making selections from a list or typing in a number or a text value of some kind.

Many requesters show a box that contains a list of items such as file names or field names. This is called a list box. The box may be only a few lines long, so there may be more files or fields than can be shown at one time. If there are more items to be viewed, you can show them by dragging down the scroll bar at the right of the box, or by clicking on the downwards pointing arrow gadget below the scroll bar.

One very important fact about requesters is that you must respond to them before you can do anything else. Most requesters have a Cancel button. Clicking this, or pressing the ESCAPE key, returns you to whatever you were doing previously. If there's no Cancel button there is certain to be an OK

button, and you can safely click this, or press the ENTER key, once you've read the requester.

Many Superbase requesters include a Clear button. This has the effect of removing anything that's in the text box, and can be used at any time.

## Using Requester Boxes

The simplest requester boxes contain just a text message and the OK and Cancel buttons. More complicated requesters may also contain a variety of other features:

- List boxes. These can show file, field or directory and drive names.
- Text boxes. When you select an item from a list box, it is automatically placed in the text box. This is also where you type in text.
- Radio buttons. If a command offers you a choice of options, each may be represented in the command's requester by an option button.
- Check boxes. Unlike option buttons, check boxes are independent of each other, so you may choose more than one at a time.
- Push Buttons. These are buttons such as OK, Clear, Cancel, Add, and Delete, which initiate an action from the requester.

Use the mouse to select items in a requester. To select an item, you point to it and click once. Clicking on the OK command button then completes the operation. When you are selecting items from a list box, you can also carry out this procedure in one step by double-clicking on the item.

If you prefer you may press the ENTER key as an alternative to clicking the OK button, and press the ESCAPE key instead of clicking the Cancel button.

## Using the Keyboard with Superbase

To avoid cumbersome repetitions, we shall usually assume from now on that you are using a mouse. Instructions, for example, will refer to clicking on the OK button rather than to pressing ENTER .

## Changing the Directory

The Open File requester is one of a number of Superbase requesters that present you with a list of file names. They all have these features in common:

- The name of the requester at the top.
- The current directory, below the requester name.
- A file name box on the left.
- A directory box on the right.
- A text box below the file and directory boxes.
- OK, Clear and Cancel command buttons.

In the text box, Superbase places a pattern matching string which restricts the files to those of a particular type. For example, the Open File requester is used for opening database files. These have SBF as their extension name, so the Text box selects them using the pattern matching string:

    *.SBF

In most file opening requesters, the file name box only shows the names of the files in the current directory that are appropriate to a particular command. The Open command is used to open database files, so its requester only lists files of this type. Likewise, the Query Open requester lists query files. If there aren't any files of the type required in the current drawer, the file name box will be empty.

**Note**    The pattern matching string cannot be edited in file opening requesters where the file names are shown without their extensions. With other requesters – such as the Export File Name requester – Superbase uses '*.*' as the pattern matching string. In this case, it is possible to edit the string and you may wish to do this in order to restrict the list of files to those with a particular extension.

When you load Superbase for the first time, the current directory is the one in which you installed Superbase. Generally, this directory is not the same as the directory or drive where you keep your data. You can check this by looking at the Open File requester. If the file name box is empty, it means that the current directory does not contain any database files.

You use the box on the right of the requester to change to a different directory. In this box, drive and directory names are shown; drive names end with a colon (':'). The directory one level up from the current directory (the parent directory) is represented by '/'.

Double-clicking on a drive or directory name in the right-hand box causes that one to become the current drive or directory, and the list of file names displayed in the left-hand box changes to show all the files of the relevant type that are now available.

Superbase allows you to open multiple files from different directories. The only restriction is that Query and Update files must be in the same directory as their associated database files. When you are opening files in other directories, the most recent directory in which you have opened a file becomes the current directory.

For the rest of this session, you will be using the **Clients** file which is supplied with Superbase as a demonstration file. Before proceeding, you need to change to the directory or drive which holds the **Clients** file.

The demonstration files are supplied on a separate disk to Superbase itself. You may have already copied them to a data directory on the hard disk. If the name of the data directory is shown, you can change to it by double clicking on its name in the directory box, or by highlighting the name and pressing ENTER .

If it does not appear in the directory box, you should first change to the parent directory by selecting /.

## Path Names

Superbase also allows you to select a file by typing its name in the requester's text box. If the file is on another drive or in another directory, you will need to give its path name. The text box acts as a scrolling window on the path name and file name. If you enter a character string longer than the window, it scrolls to the right. If you want to correct what you have typed in, you can scroll the window to the left using the LEFT key. Here are some examples of the way path names are used:

dh0:Clients

This uses the drive specifier **dh0:** to select the **Clients** file which is stored in the root directory on the disk dh0.

dh0:sb4/Clients

This selects the **Clients** file which is stored in the **sb4** directory on disk dh0.

If the directory path is shown, you may specify a pathname in any requester which is used to select files. Superbase accepts pathnames up to 66 characters long (not including the filename).

**Note**  When the drive specifier is not given, the path name you type in is appended to the path name for the current directory; for example, if you enter **sbdata/Clients** and the current directory is **dh0:sb4**, Superbase will look for the **Clients** file in **dh0:sb4/sbdata**. For this reason, you are advised always to enter the full path name including the drive specifier.

## Choosing the Clients File

After selecting Open File, change to the directory where the **Clients** file is stored. Now move the pointer to the file name box on the left. Point to the file name **Clients** and click once. Superbase copies the name of the file you've selected into the text box. (If you see the wrong file name in the box, move the pointer carefully onto the right file name, and click again.) Once you've selected the file, move the pointer onto the OK button and click once. Superbase now opens the **Clients** file. As a faster alternative to this procedure, you can double-click the name to open the file without pausing to click on OK.

## Record View

When Superbase opens a file from the start up screen, it automatically reads the first record in the file and shows it on the Work Area screen in the default View, which is Record View. You'll see the name of the file appear on the title bar at the top of the window, followed by the name of the field on which the file is currently indexed.
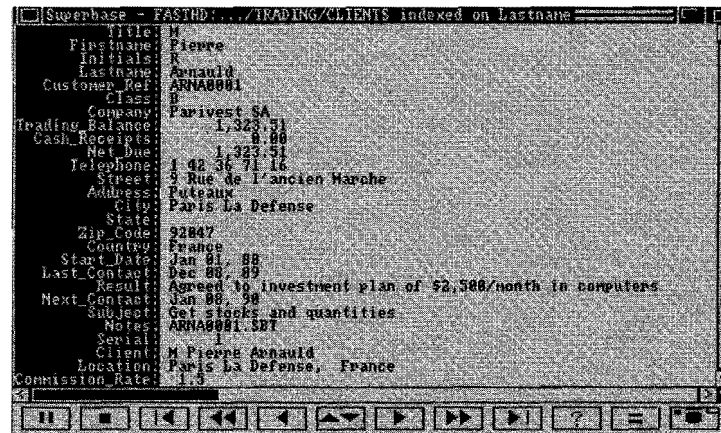
In the Work Area, the field names for each record in the **Clients** file appear down the left-hand side of the screen. The data for each field is displayed to the right of its field name. If the data for a field is wider than the window, it extends off to the right. You can bring it into view using the scroll bar at the bottom of the window.

Remember that if there were more fields in the record than you could see, you could view them by dragging the scroll bar at the right of the window downwards, or by clicking on the downward pointing arrow.

## Using the Browsing Controls

The browsing controls are a key part of Superbase's primary work surface, providing a set of tools for instant and intuitive file management.
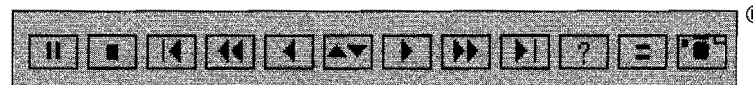


*The VCR Browsing Control design is a registered trademark of Precision Software Limited.*

You can select individual records or scan through a file with a single mouse click or keystroke. What follows here is a simple step-by-step tour of some of the browsing controls. Full details are given in Chapter 10 Retrieving Data Interactively. See Chapter 33 Command Reference Summary for a list of the keyboard equivalents to the browsing controls.

1.   Click the right arrow button to select the **next record** in the file. Repeat the action to retrieve more records.

2.   Click the left arrow button to select the **previous record**.

3.   Click the third button from the left to return to the **first record** in the file.

4.   Click its mirror image button, fourth from the right, to go to the **last record** in the file.

5. Click the double left arrow button to start a **rewind** back to the start of the file.

6. Click the double right arrow button to start a **fast forward** scan through the file.

7. Click the **pause** button, the leftmost button, to halt either a fast forward or a rewind scan. Click pause again to restart.

8. The up and down arrow button rereads the **current** record. If you have a multi-page form open, this button switches between pages.

9. The second button from the left, **stop**, ends a forward or rewind scan.

## Table View

Switch to Table View by selecting the Set Table View menu command. This places the field names across the top of the screen, and the record data in a row under them. Select the next record, using the appropriate browsing control or its keyboard equivalent. The data for the next record is displayed. You can use the same browsing controls you used for Record View with Table View.
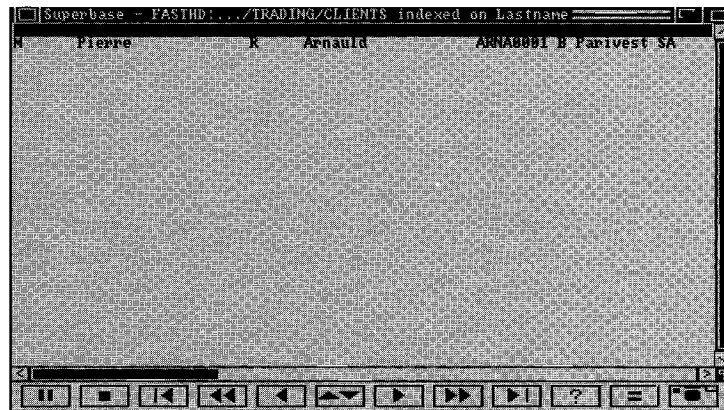


When the screen is full, the pause button is highlighted. To continue with the display, press the space bar or click the pause button.

You may move a column to the left by dragging it. Double-clicking on the column restores it to its original width.

## Page View

Now switch to Page View with the Page View command on the Set menu. The display changes to a layout in which each field is positioned independently. In this view, it is possible to have more than one field on a line. If you wish, you may move any field simply by dragging it to a new position. The only

restriction is that you cannot position one field on top of another. The browsing controls work as before.



Fields can be positioned anywhere in the 273 column by 256 row work area, so you can design page layouts to incorporate very large numbers of fields.

There is an option to switch field names off before printing, allowing you to print record data directly onto preprinted stationery.

# Using a Form

Record, Table, and Page Views provide instant access to stored information without the need for any prior screen formatting. More sophisticated views can readily be designed using the Superbase Form Designer. Let us now see a form in action by opening the Clients form supplied with your model application.

Select the Open Form command on the Project menu. The requester works in exactly the same way as the Open File requester, except that it only shows disk files in the current directory with the SBV extension, indicating that they were created with the Form Designer. A typical form might appear as follows:

Open the **Clients** form. You will see that only some of the fields in the **Clients** file are shown; this illustrates one of the benefits of forms – by selecting only the fields relevant to a particular purpose you can make it easier to read records when browsing. You may set up any number of different forms for the same file, some for browsing, some for data entry, some for printing, and of course different users of the database will require different levels of access to data.

The browsing controls work in exactly the same way when a form is open. You may step through records individually or scan them with **fast forward** or **rewind**.

If you look at the Set menu while a form is open, you will see that the Form command is now selected, whereas Table, Record, and Page View are all deselected – none of them has a checkmark. You may return to any of these views by selecting the appropriate command.

When you want to see the form again, reselect the Form command on the Set menu.

# Exiting from Superbase

When you want to leave Superbase, you can do so by selecting the Quit command on the Project menu, pressing AMIGA+Q  or by clicking on the box in the top left-hand corner of the Superbase window.

# Invoking the Superbase Form Designer

You may also run the Form Designer from within Superbase.

- Select Open Form or Modify Form from the Project menu.

    □   If a form is already open in Superbase, the Form Designer opens it ready for editing.

When you reactivate Superbase, it checks to see whether the form it has in memory has changed on disk. If it has, Superbase reloads the form.

This allows you to save the form you have been editing, and simply click in the Superbase window to use the new version of the form.

# Where To Go From Here

This chapter covers the essential information you need to begin using Superbase effectively: opening files and forms, viewing records, and using the browsing controls to review data files.

If you are new to databases, you are recommended to read Chapter 32 Learning to Use Superbase. This provides 11 tutorial lessons covering most of the basic Superbase operations. More experienced users may also want to consult the tutorials as a supplement to some of the topic descriptions given elsewhere in the manual.

As the Contents pages will show you, the rest of the manual is organized around specific database topics. Each chapter explains how to perform a particular task.

Many people will want to start creating their own databases straight away. Chapter 2, Developing Applications, outlines many of the steps you will take when setting up a database system, and provides cross-references to the relevant chapters.

The Form Designer has its own Guide, as does the programming language, DML. These are in Volume 2 of the documentation.

**Note**        You must define a file before you can create a form to use with it.

If at any time you need a quick explanation of a Superbase command, turn to Chapter 33 Command Reference Summary. This gives a brief description of each of the menu commands and also lists all the key controls and keyboard equivalents.

# 2 DEVELOPING AN APPLICATION

The scope for developing applications with Superbase is unlimited. You may have a single file of records, for which the built-in browsing controls and three default views provide sufficient management facilities. Or you may be contemplating a powerful multi-user multi-file system, with different pull-down menus for each user, and a large suite of program modules. Or perhaps your requirement calls for a specialized image database, or high quality output of forms and graphics.

Whatever your application may be, your progress through the many features that Superbase offers is likely to take a predictable course. In this chapter we present an overview of the best way for you to build a system, and point you to the appropriate areas of both the software and the documentation.

## Defining Files

The minimum requirement for a working application is a file definition. Once you have defined a new file you may add records, edit them, and delete them with menu commands. The browsing controls allow you to scan or step through your records, and to select records according to search criteria. Other menu commands provide facilities for listing and viewing data in various ways.

See Chapter 3 Defining New Files, Chapter 10 Retrieving Data Interactively, and Chapter 12 Selecting Views and Fields.

You may be using Superbase as a browsing and reporting tool for existing dBase files. See Chapter 11 Working with dBase Files.

Many applications need to use linked files. If your application requires linked files, you must analyze your data carefully and build the necessary links into your file definitions.

See Chapter 17 Multi-File Applications, and Chapter 18 Linking Files.

## Creating Input and Enquiry Forms

The next step in most applications is the creation of input and enquiry screens for use with either one or multiple files. For this you use the Form Designer module, which allows you to design many different screens, or 'forms' in Superbase terminology, for use with your files.

You may set up forms which show all or just a selection of the fields from a file. Different users will need to see different parts of records. Some forms will be designed for data entry, others for enquiry, and others for printing. Some forms may link files together so that the browsing controls can relate and retrieve associated records from several files as you step from one master record to another.

See Chapter 10 Retrieving Data Interactively, Chapter 12 Selecting Views and Fields, and Volume 2, Form Designer User Guide.

## Importing and Entering Data

Many users already have data stored in spreadsheets or other database formats. Superbase provides import options for many popular programs including Excel, Lotus 1-2-3, and dBase, as well as ASCII.

See Chapter 21 Import.

Importing data is quicker than entering it manually, but Superbase's data entry commands and data validation facilities ensure that time spent at the keyboard is efficiently used.

See Chapter 6 Entering Data, and Chapter 7 Data Entry Validation.

## Text

Many applications require the maintenance of free format text notes associated with database records, as well as the ability to generate documents for merging with data. Superbase's Text Editor provides these facilities.

See Chapter 23 Editing Text, Chapter 24 Printing Text, and Chapter 25 Mail Merge.

## Defining Queries and Reports

The next step in building an application is defining queries and reports to list and analyze your data. Menu commands allow you to create queries, and you may also generate fully formatted reports with the Form Designer. Output may be directed to screen, printer, or disk file.

See Chapter 14 Defining and Using Queries, Chapter 15 Reporting, and Volume 2, Form Designer User Guide.

## Automation: Re-usables, Macros, Function Keys

As you become more proficient and refine your database management techniques, you will begin to ask whether you can automate some of the procedures you have developed. Superbase allows you to store queries, global file updates and label formats, so that these activities can be assigned to re-usable disk files.

See Chapter 14, Defining and Using Queries, Chapter 19 Updating, and Chapter 26 Label Printing.

A further way of automating your system is to employ user-definable function keys. Again, function key commands may be stored on disk, so you may define different sets of commands for each part of your application. A function key can execute a sequence of database commands, so it is possible to perform

several actions with a single keystroke. For example, you could define a function key to change the contents of a particular field – say from 'no' to 'yes' – and store the change on disk; then you could browse through a file, pressing the function key whenever you saw a record that needed updating.

See Chapter 29 Function Keys.

## Programs

The next step for many users is to start using the program editor to combine sequences of commands into more complex procedures. For example, instead of manually opening and running five reports you could combine the five reports into one program and print all the reports with a single selection from the DML menu.

See Volume 2, DML Reference Guide.

## User Definable Pull-down Menus

Eventually, you may prefer to define your own pull-down menus to replace the default database menus for some or all parts of your application. This involves linking your own menu descriptions to specific routines or other programs that are executed when the user makes a menu selection.

# 3 DEFINING NEW FILES

## Files and Forms

Forms are the most powerful layer offered by the Superbase system for
creating database applications without the need for programming knowledge.
A form may combine fields from several different files, and may be used both
for browsing through and for entering data into multiple files.

Forms depend on files, and file definitions must be created within Superbase
before you can develop forms to use them with the Form Designer.

Many database applications are more efficient if they are designed from the
start with multiple files in mind. Chapter 17, Multi-File Applications, explains
some of the concepts that underpin the creation of such systems. Time spent
analyzing your requirements prior to defining files will be more than repaid by
not having to spend time redefining them later.

This chapter will help you set up your application whether it is a single file
address list or a complex multi-file order processing system. If you are
embarking on the development of a multi-file system, you should consult the
Form Designer User Guide for a better understanding of this higher-level tool
before proceeding to set up file definitions.

Here's a preview of the steps required to set up a new file:

1.  Select New File from the Project menu then type in a new file name,
    adding passwords if desired.

2.  Define the characteristics of each field and add it to the File Definition.

3.  Specify which fields are to be indexed.

4.  Start entering record data.

### Entering the New File Name

From the Project menu select New and with the mouse button still held down
select File. Superbase presents you with a requester for the name of the file
you are about to define.

Type the name of the new file. Don't type SBF or any other extension. It must not contain any spaces. Any characters may be used except the following:

\ = / : ;

Once you click OK, Superbase will display the Password requester. If you do not wish to set any passwords against this file, click OK. Superbase will then display the File Definition Requester which is explained below.

## Creating an Empty Copy

Some applications require the creation of new empty files with definitions that are identical to existing ones. (See Chapter 16 Reorganizing the Database). To do this:

1.  From the Project menu select Open and File, then select the file you wish to copy.

2.  From the Project menu select New and Empty Copy.

3.  Specify the name of the new file and any passwords you want it to have. Superbase creates a file which is exactly the same as the current file but without any data in it.

## Passwords

Superbase provides up to three levels of password protection. The first level, Delete, gives full access to the file, including the ability to delete data. The second level allows users Read and Write access only.

This means that they may update records, but may not delete the file or records in the file from within the Superbase system.

The third password allows Read Only access – users may not update records or delete them.

In a fully developed database, forms may be created to show only selected fields, thus protecting data from viewing as well as from unauthorized updating.

Suppose you enter the three following passwords in the requester boxes:

John
Paul
George

When you open your file and type in the password John, you will have full privileges for the file. If you enter Paul, you get read and write privileges, but you may not delete data. If you enter George you have read privilege only: no data entry or editing, and you may not delete the file or any of the records in it.

When a password exists, Superbase will ask for it each time you open the file. As a precaution against the password being seen by someone else, Superbase does not display the password on screen when you type it in.

**Caution**  Once a password has been set, it may only be modified with the DML
PASSWORD command. The file must be open.

# The File Definition Requester

Once you have entered the filename and optional passwords, the File
definition requester appears. This is used to specify the details of each field in
the file.
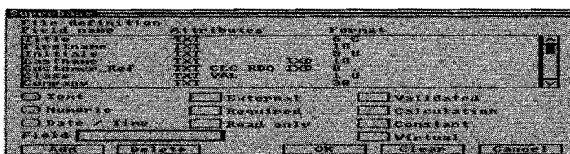


The panel at the top of the requester has three parts or zones, indicated by the
headings Field Name, Attributes and Format.

When you're looking at an existing file definition, the list of fields for the new
file appears in the left-hand zone, the abbreviated forms of the field types and
attributes (TXT – text; NUM – numeric; CLC – calculated; and so on) in the
center zone, and the field formats (length, date or time style, or number style)
in the right-hand zone.

Superbase categorizes fields into three basic types – Text/Logical, Numeric,
and Date/Time. Each of these is represented by a radio button below the panel
on the left. For each basic type there are a number of format options. The
requesters for specifying or amending formats are invoked by clicking the
relevant radio button. The check boxes to the right of the radio buttons refer
to attributes. The requesters for specifying or amending attributes may be
invoked by clicking the relevant check box. Attributes may apply singly or in
combination to any of the field types, subject to certain rules. For example, the
External attribute always refers to a Text field, Virtual fields are always Read
Only, and so on. Detailed descriptions of the formats and attributes are given
below.

At the foot of the File definition requester and to the left are the Add, Delete,
and Name buttons. When you've defined the details of a new field, you click
Add, and Superbase adds the field name together with its attributes and

format to the upper panel. Delete is used when editing an existing file definition.

The Name button is used to change the name of a field after it has been added to the file definition. If you wish to change a field name, the procedure is:

1.   Click on the name in the File Definition panel. Superbase then places the name in the Field box.

2.   Click in the Field box, delete or edit the existing name, and type in a new name.

3.   Click the Name button.

At the foot of the requester and to the right are the usual OK, Clear and Cancel buttons. If you click Cancel, Superbase first asks if you want to proceed with this action and then exits from the requester to the main display. Clear empties the Field box and resets the field type selection buttons. OK takes you on to the next step, which for a new file is specifying the index fields.

To summarize, the procedure for defining new fields is as follows:

1.   Type in a field name.

2.   Select the appropriate field type button.

3.   Enter details about the field – its length or numeric format, for example.

4.   Use the Add button to add the field to the file definition.

# Creating a New Field

The first step in creating a new field is to type in the field name. Field names are important. They are used in views and as the default headings for query output. They may be used as variables in DML programs. Ideally they should be short but descriptive. There is no reason why you should not use the same name, for example **Desc**, to refer to fields in different files. When defining formulas and command lines, Superbase automatically appends the file name to the field name, creating a unique field name such as **Desc.INVENTRY**.

Now, click in the Field box below the Date/Time field type button. Type in the field name. You may use up to 15 characters, including spaces. *Use only alphanumeric characters, the space, and the underline character. The field name may not begin with a number or contain more than one consecutive space.* You can use any words except those reserved for the Superbase programming language. A list of these is given in Appendix C.

## Selecting the Field Type

Click the button next to the type of field you want. Choose from Text/Logical, Numeric, Date/Time. If you wish to create an External File field, click the Text/Logical button and the External check box.

You can add attributes such as Validation or Read Only later. Your choice will display another requester for entry of details about the field. (Each of the field types has its own requester. These are described in the sections following this one). When you have specified the details, you click OK in the field type requester to return to the File Definition requester.

# Text/Logical Field

Use this type for any field that is not designed to hold a date, a time, or a number.

The Logical field type is a sub-type of the Text field type. Logical fields contain a limited range of values, making it easier to predict their contents when specifying search criteria.

Defining a text field involves specifying its length, its attributes, and whether it is to be a multiple response field or not:

1.  Click the Text button. Superbase displays the Text requester.
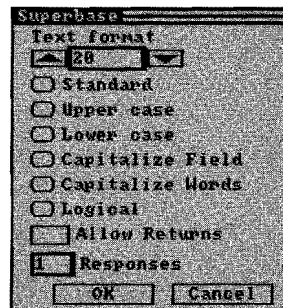


2.  Set the length of the text field, up to a maximum of 4000 characters, by entering it in the box at the top of the requester. There are two ways of doing this.

    □ From the keyboard. Click in the box to activate the insertion point, then enter the number of characters from the keyboard.

    □ Using the mouse. Set the length using the right and left arrows at either side of the box. The right arrow increases the value in the length box, the left arrow decreases it. The value in the counter changes each time you click one of the arrows.

3.  Set the text attributes by clicking one of the five first option buttons. These options determine how text is stored in a field. If you leave the default, Standard, selected, the text will be stored and displayed in the same case in which it is entered (standard text format).

4.  Click the Logical field button if you wish to define a logical field. The field will have a fixed length of one character, and you will only be able to enter 'Y', 'N', 'T', or 'F' into it. Logical fields are distinguished from others in the File Definition panel by the LOG type abbreviation.

5.  If you want the field to include Returns, so that you can format it with paragraphs and blank lines, turn on the Allow Returns check box. If this attribute is included, the File Definition Panel shows the letter R next to the field length.

6.  The box labeled Responses is used to define a text field as a multiple response field. This attribute, which is unique to text fields, is explained in the last section in this chapter. Normally the box contains the number '1' which indicates that the text field has has only one response. Unless you want the text field to be multiple response, you can leave this box as it is.

7.  When you have set the text format, click OK to accept the entry.

# Numeric Field

Use a numeric field for all numbers, including decimal numbers, currency amounts, integers, percentages or numbers expressed in exponential format. Superbase stores numbers with a maximum precision of 14 digits, which you may arrange in any format around the decimal point.

When you define a numeric field you control two quite separate aspects of the numbers that will be stored in the field:

• the way the numbers are stored by Superbase (controlled by your selection of real, long or integer in the requester described below), and

• the presentation of the numbers when they are displayed or printed (controlled by all the other options in the requester).

1.  Click the Numeric button. Superbase displays the numeric field requester.
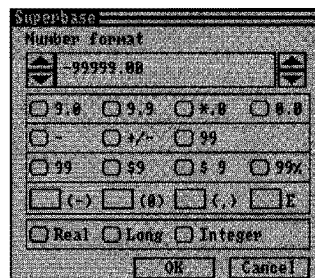
2. When the requester first appears, the number format is set to -99999.00. This allows up to five digits to the left of the decimal point, and two decimal places. There is also a space for a minus sign for negative numbers. To change the numeric format length, click the arrows at either side of the number format panel. The left-hand arrows increase and decrease the number of digits to the left of the decimal point, the right-hand arrows increase and decrease the number of decimal places. It is not possible to edit the number format directly.

3. Select the numeric format type and other numeric features, as described in the following sections.

4. Click OK.

## Rounding

The numeric format only determines how numeric field data is displayed; it does not affect the way numbers are stored (Superbase stores real numbers to 14 figure accuracy). For example, if you set the format to three decimal places and type in the number to five decimal places such as 5.66666, it will be displayed as:

5.667

To store the number 5.66666 as 5.667 you would use the FIX function. See the entry for FIX in Volume 2, DML Reference Guide. The STR$ and VAL functions may also be used to round numbers.

## Money Format

The exception to the above is the numeric format with two decimal places. Superbase treats numbers using this format as money amounts. Such numbers are both displayed and stored to 2 decimal place accuracy; so if you entered 5.66666, it would be stored as 5.67.

## Selecting Numeric Format Type

You can choose from four possible types of basic numeric format; the selection is always reflected in the number format panel:

**9.0**   This format is preselected when the requester appears. The format shows actual numbers left of the decimal point, and trailing zeros to fill up the decimal places. For a format of 999.000:

26.200
334.456

**9.9**   This format shows numbers only, with leading or trailing spaces:

26.2
334.456

**\*.0**                    Inserts leading asterisks in place of leading zeros; adds trailing
                            zeros to the right. For a format of \*\*\*\*\*.00:

                            ```
                            ***26.20
                            **334.46
                            ```

**0.0**                     Inserts both leading and trailing zeros. For a format of 0000.000:

                            ```
                            0026.200
                            0334.456
                            ```

## Selecting Numeric Format Features

You may choose from three number sign styles:

**-99**                     This button is preselected. It causes Superbase to print a minus
                            sign if the number is negative, a space if the number is positive.
                            Shown as - in the number format requester box.

**+/-99**                   Prints a minus sign if the number is negative, a plus sign if the
                            number is positive. Shown as + in the number format requester
                            box.

**99**                      This means that Superbase does not leave a space for the sign in
                            the format. It does, however, insert a minus sign when the number
                            is actually negative. If you enter negative numbers into a field
                            which has no provision for a sign, they will not be properly
                            displayed in column output, nor when the field is edited.

There are three currency sign options:

**99**                      No currency sign.

**$99**                     Places the currency sign immediately to the left of the first digit in
                            the format:

                            ```
                            $26.20
                            $334.46
                            ```

                            Shown as $9 in the number format requester box.

**$ 99**                    The currency sign is left-aligned – it appears to the left of the
                            format, always in the same place. For a format of $ 9999.00:

                            ```
                            $       26.20
                            $       334.46
                            ```

                            Plus and minus signs are also aligned when this style is selected.

You may choose your preferred currency symbol by typing it in the Currency box in the Set System Options requester. This also allows you to set a currency suffix for the currency sign style, as in:

26.20$

or:

26.20£

This is a global setting and may not be varied from field to field.

As an alternative to a currency symbol, the percent symbol may be selected. All numbers using this format will appear with a trailing percent symbol.

26.20%

The check boxes at the bottom of the requester provide four other options affecting numeric formats:

(-) Prints parentheses instead of the minus sign. Can be selected in conjunction with one of the above. Shown as '(' in the number format requester box.

(0) Zero suppression. This makes Superbase print blank spaces instead of zeros when the number to be printed equals zero. Shown as z in the number format requester box.

(,) Thousands separator. Numbers greater than 999 are printed with a separator, e.g. 1,100. Normally the separator is a comma and the decimal point is a period or full stop, but alternatives may be selected from the Set Options requester. Shown as a comma in the number format requester box.

E Exponential format. All numbers whose format includes an 'e' will be expressed with an exponential signifier at the end. This allows you to enter numbers much larger than the normal 14 digit precision permits. Shown as 'e' at the right of the number format requester box.

## Overflow

If the number entered into a numeric field during data entry is too big for the format, Superbase displays number signs to indicate an overflow condition:

########.##

You can easily edit the numeric format to make it longer, or, if you prefer, include an exponential signifier in the format.

### Internal Storage

You may choose between real, long or integer. Your choice is determined by the type of numeric values to be stored in the field, and by considerations of amount of storage occupied and speed of operation.

'Real' numeric fields can hold any numeric value, but occupy most storage space and are slowest in operation. 'Long' numeric fields can hold only integers (from –2,147,483,648 to +2,147,483,647) and are moderate in space requirement and speed. 'Integer' numeric fields can hold only integers (from –32,768 to +32,767), occupy the minimum space and are quickest in operation.

Typically, therefore, you would use a 'real' numeric field for amounts of currency, for percentages or for the results of an arithmetic calculation. You would use a 'long' numeric field for counters that may take large values, such as numbers of votes cast, documents issued, items sold and so on. You should use 'integer' numeric fields whenever possible, especially for fields you intend to use as indexes to database files. The space required by index files, and the time taken by Superbase to read them, is then minimized.

# Date Fields

Date fields can accept any date (years 0001 to 9999). There are many possible styles of date expression.
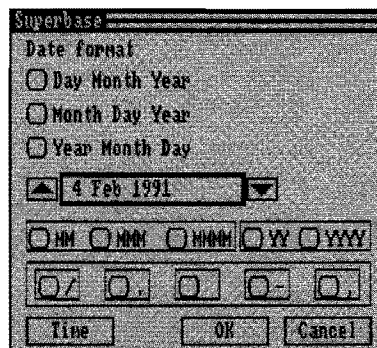


When the date field requester appears, the date is in the default date style. You can reset it by altering the style with the Set Date Format menu command.

### Changing the System Date

The box in the middle of the requester shows the current Superbase system date. To change the date, click the left arrow to increase it, or the right arrow to decrease it. Superbase remembers the new date whether you set it from here or with the Set Date command.

**Caution**  Other programs will also use this date as their system date.

You can also edit the date directly by clicking in the box.

The system date always reflects the current date style as determined by your selections.

## Day Month Year Order

Click one of the buttons in the upper part of the requester to set the order in which the day, month and year are presented.

## Month Style

You can choose one of three possible month styles:

**MM**      Number from 1 to 12.

**MMM**    Three letter abbreviation: Jan.

**MMMM**  Full name of month: January.

## Year Style

You can choose one of two possible year styles:

**YY**      Last two digits of the year: 91. This always implies the current century.

**YYYY**   The full year: 1991.

## Separators

There are five kinds of separator to choose from.

/              1/10/91

-              1-10-91

**space**    1 10 91

.              1.10.91

,              First separator is space, second is a comma followed by a space: 1 October, 1991 or Oct 1, 1991.

## Dates As Numbers

Although dates are expressed as text, they are stored internally as numbers. This means you can include them in arithmetic expressions, and incorporate them in formulas and validation checks. You can also construct an index on a date field knowing that it will be kept in true date order.

# Time Fields
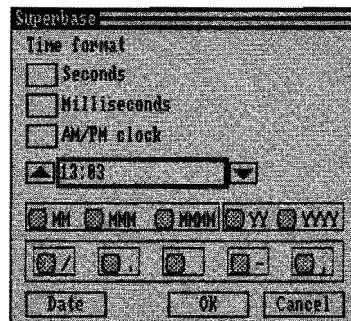


Time fields offer a variety of formats for storing the time of day, or for storing any elapsed time period up to 24 hours. You can choose between a 12 or 24 hour format and there are three ways of specifying the time units:

**hours and minutes**
**hours, minutes and seconds**
**hours, minutes, seconds and thousandths of a second**

You set the time format by clicking on the Time button at the bottom of the Time/Date requester. The selection buttons will then present the various time format options: seconds, milliseconds, and am/pm. The default format shows the time in hours and minutes on a 24 hour clock. Click the selection buttons to set this to another format.

Like dates, time fields are expressed as text, but are stored internally as numbers.

## Changing the System Time

The box in the middle of the requester shows the system time in the currently selected time format. As with the system date, you can change the system time by clicking on the arrows at the side of the box, or by editing the box directly.

# External File Fields

An external file field is a special type of text field, intended to contain a file name.

To create an external file field, click the Text button and define the field in the same way as you would define a normal text field. Then click the External check box.

The file names in external file fields are used in Superbase's External File System, which allows you to find and display graphic images or text files. It provides both a kind of built-in 'picture library' facility, and a 'memo fields' facility. See Chapter 28 Integrating Image and Text Files.

# Adding Fields

After setting the details for a field type, the next step is to add it to the list of fields in the Field Names box. Click the Add button in the bottom left-hand corner of the requester. Superbase then:

1. Copies the name from the Field box into the File Definition Panel, adding it to the existing list of names.

2. Shows abbreviations to indicate the field type selections and the field attributes (both are shown under the heading Attributes). These appear across four columns, like this:

TXT          VAL          REQ          IXD
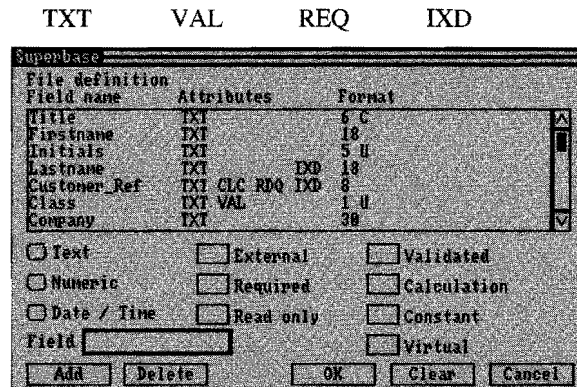


The abbreviations for the field types and attributes are as follows:

**First column**
TXT  Text
LOG  Logical
NMI  Numeric - Integer
NML  Numeric - Long
NUM  Numeric - Real
DAT  Date
TIM  Time
EXT  External
DEL  Deleted

**Second column**
CLC  Calculated
CLV  Calculated and Validated
CON  Constant
COV  Constant and Validated
VAL  Validated

**Third column**
RDO  Read Only

REQ   Required
RDQ   Required and Read Only

**Fourth column**
IXD   Duplicates Index
IXU   Unique Index

**Note**     Indexes are explained in Chapter 4, Creating Indexes.

To show that a field is virtual, Superbase takes the first two letters of its
type abbreviation, and precedes it with the letter V. For example:

VTX        (for virtual text fields)
VNU        (for virtual numeric fields)
VDA        (for virtual date fields)

3.  Shows the format of the field. For a text field this is the length of the field
    in characters. It may also include the two optional text parameters: case
    parameters (Upper Case, Lower Case, Capitalize Field, Capitalize
    Words) represented as U, L, C and W; and the number of multiple
    responses selected (if more than one) represented as M followed by the
    number. The character L indicates that the length is greater than 255
    characters.

    For a numeric field the format is specified by the digit pattern plus certain
    characters to show which style features have been included.

    For a date field the format is a representation of the day, month and year
    style you selected, and for a time field the format is a representation of the
    selected time style.

    Validation and Calculation formulas are not shown. You may, however,
    view or print them with the Utilities Status File command.

4.  Finally, Superbase clears the Field box and the type selection buttons.
    You can either define another field or exit from the definition requester.

## Other Data Attributes

Each field in a Superbase file is one of the six types described above: text,
logical, numeric, date, time or external. When you look at the File Definition
requester, you will see that there are six more field selection options:
Required, Read Only, Validation, Calculation, Constant, and Virtual.

### Required

You may designate any field as a required field. This means that during data
entry you must enter a value – you cannot leave a required field blank.
Required fields are checked when you exit from them (and also when the
record is saved). If you press the ENTER key in a required field without

entering any data, Superbase displays a requester with the message 'This field must have some data.'

## Read Only Fields

The Read Only attribute is used to protect the data in the field from being deleted or overwritten. Data cannot be entered in a Read Only field.

Although any field may be designated as Read Only, normally you would only use this attribute for fields which have Constant or Calculation formulas attached to them. If you designate any other kind of field as Read Only, it will not be possible to enter data in the field, and, as a result, it will remain blank. However you may want to protect data after it has been entered in a field. In this case, you would change the field's attribute to Read Only by modifying the file definition.

## Virtual Fields

Virtual fields provide a way of saving disk space. Any field which is defined as virtual must have a Constant or Calculation formula attached to it. When you save a record, Superbase calculates the value of the virtual field, and if it is a key field, creates an index pointer for the record on the basis of its derived value.

However, the value is not stored with the record. Instead, Superbase uses just one byte (one character) per record to signify a virtual field. The value is then recalculated whenever it is required:

- When you display the record.
- When you edit the record.
- When a LOOKUP formula, which specifies the virtual field, is performed in another file.

When you are browsing through a file, Superbase will recalculate virtual fields as it retrieves them. With complex calculation formulas, this may have a slight effect on the program's performance.

## Multiple Response Fields

One way of explaining about fields is to say that each field holds the answer to a specific question about the data in a record – such as, what is this client's telephone number? Or, how much does this product cost? As the name suggests, multiple response fields are used when there may be more than one answer to the question. Suppose you wanted to index the books in a library. For each book you would create a separate record. Along with the title and other details, the record would contain a field showing the author of the book. Some books, though, have more than one author, so you could make this field a multiple response field.

Defining a text field as a multiple response field enables you to store a number of related data items under one field name. When you display a record with a multiple response field, only one of the field's items is shown on screen; but you can display each item in turn using the key combination CTRL+N . To move back to a previous item, use CTRL+P . For example, if the second is currently displayed, CTRL+N displays the third item; CTRL+P will then return you to the second item.

There are some constraints on the use of multiple response fields. References to the individual elements of a multiple response field must be by subscript – entering the number in parentheses after the fieldname – and the reference must always be typed in. Typical examples are when setting up a filter or in a calculation formula linking one field to another. The number for the first item is zero, so to refer to the third item in a multiple field, you would enter:

Fieldname(2)

Some multiple response field applications, such as labels, are best served by being fully programmed in DML. Note also that multiple response fields are not supported in Virtual Field Indexes, or in Transaction Lines defined on a Form.

### Defining a Multiple Response Field

Multiple response fields may have up to ten elements; that is, you can use a multiple response field to store up to ten separate lines of text. The total number of characters in all the elements must not be more than 4000.

To define a text field as a multiple response field:

1.  Decide how many elements you need.

2.  Enter the number in the box labeled Responses in the text field requester. Initially, the number will be set to one. Delete this using the backspace key and type in the new number.

# Saving the File Definition

When you have defined all the fields you require, click the OK button in the File Definition requester. Superbase will then present you with the New Index requester. This stage in the process of defining a new file is described in Chapter 4 Creating Indexes.

Once you have specified the fields that are to be indexed, Superbase automatically completes the file definition process and displays a blank record for the new file. You can now start entering data. See Chapter 6 Entering Data for more information.

If you wish to print out your file definition, set up your printer, select the Print command on the Project menu, and then choose the Status File option. See Chapter 31 Housekeeping, for more information.

# 4 CREATING INDEXES

## Overview

A Superbase index is a separate file that is related to the main data file. It corresponds to one of the fields in the File Definition. The index keeps a copy of all the data you enter into that field, stored in alphabetical order (actually spaces, punctuation marks and numbers come before the alphabet).

An index file on disk has a name of the form 'filename.nnn', where 'filename' is the same as for the data file to which the index is related, and 'nnn' is a number corresponding to the position of the indexed field in the File Definition's list of fields. Example: 'Clients.1'

You may have up to 999 indexes for each file, using any of the first 999 fields in the File Definition. All indexes are automatically updated when you add records to the file. If you create a large number of indexes, updating them will obviously take longer than if there are only a few indexes to process.

## Using Indexes

An index is used in two ways: to provide a quick method of looking up individual records, and to provide a sequence for the presentation of records.

A file's indexes are all opened automatically when you open the file. (This is only possible for a dBase file if you have previously opened all the relevant indexes and then used the File Save command to create a DBD file.)

The default order for an index is ascending: A-Z; you may, however, reverse the default order for the current session by selecting Open and Index from the Project menu and checking the Descending order check box.

Use the Key Lookup button on the browsing controls to enter an index value when you want to retrieve a single record by key. Use the Index Open command when you want to switch the order of presentation from one field to another, or press the '-' key on the numeric keypad to cycle through each of the file's indexes in turn.

# Unique Index

A unique index is one which does not allow duplicate entries: the data in a field must not be repeated in any other record. Unique index fields are checked when you exit from them after entering data and also when the record is saved. If you enter data which is duplicated in another record, Superbase displays a requester with the error message 'Duplicate entry for this index.'

If you are adding an index to an existing File Definition, you select a field for indexing using the Index New requester. To create a unique index:

1.  Select New and Index from the Project menu. Superbase displays a requester showing the names of the fields in the current file that are not already indexed (see next page).

2.  Click the unique index button.

3.  In the list box showing the field names, click the field to be indexed and then click OK, or just double-click the field.

4.  Superbase now shows a warning message telling you that creating a new index may take some time. Click OK to proceed.

For more information on creating unique indexes, see **Generating Serial Numbers**, Chapter 8 Derived Values.

# Normal or Duplicate Index

Use a duplicate index for fields which contain multiple occurrences of the same data: indexes of this kind allow duplicate entries. For example, a clients file would probably contain many occurrences of the same last name; so if you wanted to index the Lastname field, you would use a duplicate index.

To create a duplicate index, follow the same steps as for unique indexes, but for Step 2 click the normal index button.

# Indexing a New File

When you're creating a new file, you are forced to pass through the Index New requester, as every Superbase file must have at least one index. The requester appears when you click OK in the File Definition requester:

1.  Click on the name of the field on which you want the file to be indexed. Superbase copies it into the Text box.

2.  Decide whether the index is to contain unique entries only or to permit duplicate entries. Click on the appropriate button.

3.  Superbase displays the Index New requester again, allowing you to select other fields for indexing. If you want your file to be indexed on more than one field, repeat the procedure described above.

4.  To indicate that you have completed index selection, click OK without entering a field name in the text box.

This is the last stage in the process of defining a new file. Superbase now creates all the disk files that comprise the database file: the SBF extension file for the record data, the SBD file for the definition, and the *.nnn* files for the

indexes. *Note that you should create an index for any field which will be used for relational purposes.*

# Composite Indexes

A composite index applies to more than one field at a time. To create a composite index, you need to define a virtual field which will concatenate the fields you wish to index:

1.  In the File Definition requester, type in a name for the composite index field.

2.  Define the field as a virtual text field. In the Text Format requester, set the length to the sum of the lengths of fields that will be concatenated; i.e., if you are creating a composite index on two fields that are 20 and 30 characters long, the composite index field should be 50 characters long.

3.  Click the Calculation check box and type in a calculation formula to concatenate the fields which are to be indexed. Examples of suitable calculation formulas are:

    > Country.Clients + City.Clients
    > PAD$(Lastname.Invoice)+STR$(DAYS(Date.Invoice,"0000000"))

The first formula would be used to create a composite index on the **Country** and **City** fields. With the second formula, the file would be indexed first in the order of the **Lastname** field, then according to the **Date** field. DAYS is a Superbase function which converts the date value to a julian day number; the STR$ function then converts the date value (now expressed as a number) into a text string. For an explanation of the PAD$ function, see below.

4.  Click the Add button.

5.  Click OK.

6.  Set an index on the field, using the New Index requester.

When you are defining a calculation formula to concatenate fields, it may be necessary to pad out the first field with spaces using the PAD$ function. PAD$ checks the length of the field in the file definition and then adds spaces to the field data so that it has the same number of characters as in the file definition.

For example, if the **Lastname** field in the **Clients** file has been defined as 20 characters long, and the name in this field is 'Richardson':

> PAD$(Lastname.Clients)

generates the text string:

> "Richardson          "

i.e., it adds 10 spaces to the name.

Why is it sometimes necessary to use the PAD$ function? A simple example should make this clear.

Imagine you have a large **Clients** file in which many last names are duplicated. Thus there may be dozens of people who share the name 'Smith' or 'Jones'.

As it stands, there is no simple way of selecting the record for, say, 'Michael Smith' before 'Adam Smith'. Yet, for some applications, you might need to sort or retrieve duplicate last names in the order of the **Firstname** field.

The solution is to create a composite index on the data in both fields, **Lastname** and **Firstname**. However, it is not enough simply to concatenate the two fields using the calculation formula:

Lastname.Clients + Firstname.Clients

The reason is that there may be other individuals in the file whose names contain 'Smith' as a prefix – John Smithers and Alice Smithson, for example. When the last and first names have been concatenated, the order within the composite index field will be:

SmithAdam
SmithersJohn
SmithMichael
SmithsonAlice

As you can see, this sequence ignores the alphabetical order of both the Lastname and Firstname fields.

If instead the calculation formula is:

PAD$(Lastname.Clients) + Firstname.Clients

these records would be indexed on the concatenated field as follows:

| | |
|---|---|
| Smith | Adam |
| Smith | Michael |
| Smithers | John |
| Smithson | Alice |

In alphabetical order, the space character takes precedence over a letter or a number, so the composite index works as intended.

**Note**      Composite index fields must be defined as text fields. If you wish to concatenate a numeric field, you must first convert it to text format using the STR$ function. Date fields can be converted to text using the DATE$ function, but as text they cannot be indexed according to their temporal order; i.e., '5 JULY 1989' comes before '7 JULY 1988' in alphabetical order. Instead of DATE$ use an expression such as STR$(DAYS(datefield)), which converts the date value to a julian date number.

## Example

Suppose you wanted to display the records in a clients file in alphabetical
order according to the **Country** field. Within each country, you might also want
the records to be displayed in the order of the **Lastname** field. The way to do
this is to create an indexed virtual field with the calculation formula:

    PAD$(Country.Clients) + Lastname.Clients

Now when you select the virtual field as the current index, it indexes the
records at two levels: first according to the **Country** field, second according to
the **Lastname** field. In a large file, if you were to concatenate **Lastname** and
**Country** in a non-virtual field, their data would occupy a substantial amount of
disk space. A virtual field, however, only takes up one byte extra for each
record in the file. See also Chapter 8 Derived Values.
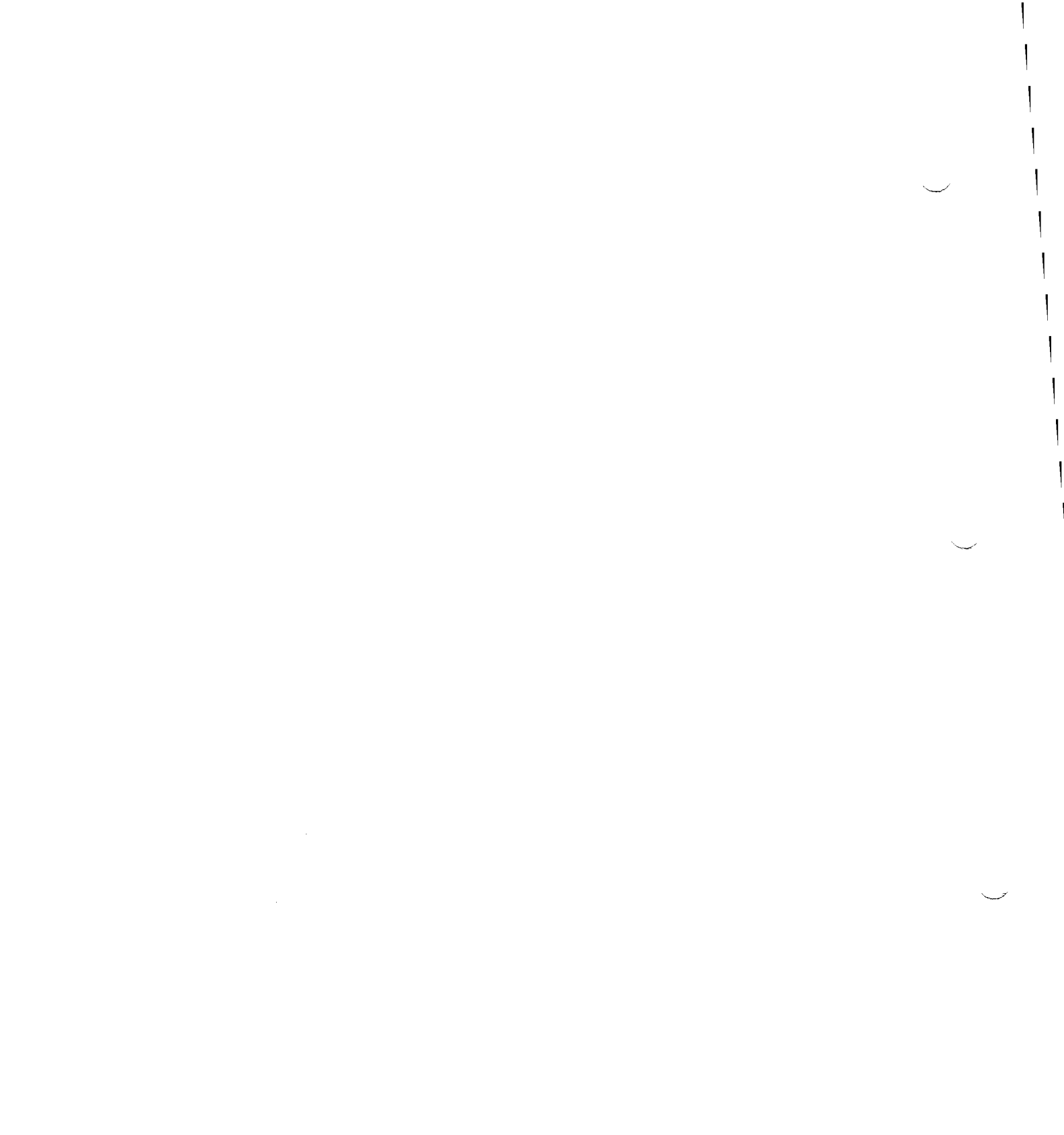
# 5 MODIFYING A FILE

However careful you are when you design and create a new file, there inevitably comes a time when you want to change some details. Maybe you made a text field too short, or you find overflow characters appearing when you try to enter data into a numeric field. Or perhaps you decide that a certain field needs a range check adding to ensure the integrity of the data in it. Superbase lets you make almost any change to the original File Definition, and carry on using the file without any delay. You even have a choice which allows you to use the File Definition settings experimentally until you're sure they are what you want, before saving them permanently.

Here's a summary of possible changes to a File Definition. You can:

* Change a field's characteristics. These include: its numeric format, length, date or time style; its validation, calculation or constant formulas; and its attributes (Read only, Required and Virtual).

* Add new fields to the end of the list of fields.

* Delete a field.

* Rename a field.

* Change a field's type; e.g., from text to numeric.

The first steps in changing a field are:

1. Open to file you want to change – in the Project menu, point to Open and File, then select the required file.

2. In the Project menu, point to Modify and File - the File Definition requester is displayed.

## Changing Field Characteristics

The basic procedure for changing a field's characteristics is similar to that for creating a new file.

1. Click on the name of the field you want to change. Superbase copies its name into the Field box, and selects the buttons that correspond to the field's attributes.

2. Change the field name if you want to. Use the backspace key to delete the existing name, type in the new name, and click the Name button.

3. Click on an attribute button to bring up a Field Type requester so you can alter the details of the field. You can choose a different type from the existing type.

4. When you click on OK in the Field Type requester, Superbase updates the File Definition Panel with any changes you made.

5. If a field is an indexed field, you will have to remove the index before you can change its field type.

6. When you've finished making changes to fields, click OK in the File Definition requester itself. Superbase asks you whether you want to save the modified definition. You also have the option to try it out in memory, either saving it or abandoning it later.

**Note**    If you add a calculation to a field or you modify an existing calculation, you will probably want to recalculate the field value in all the records in the file. You can do this in a single operation using the Update command. See **Forcing Calculations to Evaluate**, Chapter 19 Updating.

## Adding a New Field

Do exactly what you would do when creating a field with the File New command:

1. Ensure the Field box is empty – click Clear if necessary.

2. Type in the new field name.

3. Click the appropriate attribute button and add the required details in the Field Type requester.

4. Click Add and the field name will appear at the end of the existing list.

If you add new fields to an existing file definition, they will be entered at the bottom of the field list. It is not possible to insert new fields between existing fields using the Modify command. However, you can change the order of a file's fields by creating a copy of the file using the Query Output to File option. See Chapter 16 Reorganizing the Database for more information.

## Deleting a Field

1. Click the name of the field to be deleted.

2. Click the Delete button.

When you do this, Superbase places the DEL abbreviation in the Attributes column of the File Definition Panel. The field will no longer be available when you retrieve records from the file. However, Superbase leaves the field data in place in the SBF file, and leaves the field name in place in the File Definition, until either:

•   You tidy up the file with the Process Reorganize menu command. See Chapter 16 Reorganizing the Database for more information.

or:

- You modify a record and save it; for example, with the Process Update menu command. In this case Superbase removes the original data from the file for the records it modifies. See Chapter 19 Updating for more information.

Because of the delay between marking a field as deleted and actually removing the data, you can recover deleted fields by clicking on them and selecting the appropriate Attribute type button, as for a new field.

### Renaming a Field

1. Click the name of the field you want to rename. Superbase copies its name into the Field box, and selects the check boxes that correspond to the field's attributes.

2. Use the backspace key to delete the existing name and type in the new name.

3. Click the Name button.

### Changing Field Types

1. Click the name of the field you want to change.

2. Click the radio button for a different field type from the existing type.

3. Using the field type Format requester, set the format for the new type.

4. Click OK in the Format requester. Superbase updates the File Definition with any changes you made.

# Saving the Modified File Definition

When you have finished making changes to your File Definition, you must click OK to put the modified definition into effect. Superbase displays the Save File Definition requester.

Click OK to store the File Definition.

**Note**     If you click Cancel, you can continue to work with the settings as they are, without making any permanent change to the SBD file. This allows you to experiment freely with the details of the file definition, testing out various settings and formulas until you're sure you've got them right. When you're satisfied, you simply select the Save File command on the Project menu and click OK when the requester appears.

# Rules for Changing Fields

## Text / Logical

Length may be varied freely.

- Validation, Calculation or Constant formulas may be added, changed or removed.
- Required status may be added or removed.
- Read Only may be turned on or off.
- Multiple response fields may be added or removed
- Type may be changed to numeric, date, time, external or virtual:
  - ☐ If changed to numeric, all text will be lost when the field is next saved.
  - ☐ If changed to date, all non-date text will be lost when the field is next saved.
  - ☐ If changed to time, all text in the field will be lost when it is saved.
  - ☐ If changed to external, Superbase will not be able to find non-existent external files, and will reject any text that would be an illegal file pathname.
  - ☐ If changed to virtual, all text in the field will be lost when it is next saved.

## Numeric

Number pattern and features may be varied freely.

- Validation, Calculation or Constant formulas may be added, changed or removed.
- Read Only may be turned on or off.
- Required status may be added or removed.
- Type may be changed to text, date, time, external or virtual:
  - ☐ If changed to text, the number will become a character string.
  - ☐ If changed to date, the number will be treated as a julian date and interpreted according to the style you choose.
  - ☐ If changed to time, the number will be treated as thousandths of a second and interpreted according to the time format you choose.
  - ☐ If changed to external, Superbase will not be able to find non-existent external files, and will reject any text that would be an illegal file pathname.
  - ☐ If changed to virtual, the number will be lost when the field is next saved.

## Date

Date styles may be varied freely.

- Validation, Calculation, or Constant formulas may be added, changed or removed.
- Required status may be added or removed.
- Type may be changed to text, numeric, time, external or virtual:
  - ☐ If changed to text, the date will become a character string of the julian date, i.e. a string of numbers.
  - ☐ If changed to numeric, the date will be displayed as a number according to the numeric format you select.
  - ☐ If changed to time, the date's julian date number will be treated as thousandths of a second.
  - ☐ If changed to external, Superbase will not be able to find non-existent external files, and will reject any text that would be an illegal file pathname.
  - ☐ If changed to virtual, the date will be lost when the field is next saved.

## Time

Time formats may be varied freely.

- Validation, Calculation, or Constant formulas may be added, changed or removed.
- Required status may be added or removed.
- Type may be changed to text, numeric, date, external or virtual:
  - ☐ If changed to text, the time will be shown in thousandths of a second as a character string.
  - ☐ If changed to numeric, the time will be displayed as the number of thousandths of a second.
  - ☐ If changed to date, the time in thousandths of a second will be treated as a julian date number.
  - ☐ If changed to external, Superbase will not be able to find non-existent external files, and will reject any text that would be an illegal file pathname.
  - ☐ If changed to virtual, the number will be lost when the field is next saved.

### Validation, Calculation and Constant

The details of the formula may be varied freely.

- The formula may be removed by clicking on Clear, then OK.
- Required status may be added or removed.

**Note**   Direct editing of the formula may make it invalid.

### Read Only

May be added or removed.

### Required

May be added or removed.

### Virtual

May be added or removed.

**Note**   If you change the calculation for a virtual field, the index for that field will not be updated and will not work as expected. The correct procedure is to remove the index first, edit the calculation, and then rebuild the index.

# 6 ENTERING DATA

You may enter and edit data through either a form or the Page or Record default views. This chapter covers both situations. Page and Record View operate on one file at a time, whereas forms may include fields from many files, and can update all of them with a single Save command.

There are three ways of entering data into a file: interactively using the Record and Edit menus, from the Clipboard, or through a batch import. In this chapter we deal with the first two methods. See Chapter 21 Import for information on importing data from another file.

## Entering New Records in Page or Record View

Each record in a file has the same format, as set out in the file definition, consisting of a number of fields of different types. If it helps, you can imagine each record as a card in a card index, or a form in a folder.

When you want to create a new record ('add a card to the box,' or 'add a form to the folder'), you have to follow a simple procedure:

1. In the Project menu, point to Open and File, then select the required file.

2. In the Record menu, point to New (or press AMIGA+N ). Superbase presents a blank record with the insertion point in the first editable field. A blank record for the example **Clients** file would look like this.



   □ If there are any constant fields, their initial values are displayed.

3. Type in the data for the first field, pressing ENTER to finish.

   □ You can exit from data entry at any time by pressing ESCAPE .

   □ You can save a record at any time with the AMIGA+S shortcut provided the insertion point is not active.

□ You must use TAB to exit from a text field that allows Returns. See below.

4.  Repeat step 3 for each field.

□ After you press ENTER in the last field, Superbase asks you whether you want to save the record (add it to the file).

5.  Press ENTER to save the record, or ESCAPE to stop.

□ Whenever you save a record Superbase presents you with a blank record so that you can create another. Press ESCAPE if you do not want to create another record.

If you want to duplicate an existing record, display it then point to Duplicate in the Record menu.

If you make mistakes while typing in data, you can correct them using the Backspace and Delete keys together with the other editing controls explained below.

If you have set up validation formulas in the file definition, these are performed as you enter data. Required fields may not be left blank.

Unique index key entries are checked when you press ENTER in the key field. You may leave an index field blank – Superbase makes an entry of spaces in the index, and the record will appear first in sorting order. Remember that if the index only allows unique entries it will not let you store more than one record with a blank unique index field.

## Entering Text

When you edit a text field whose entire length is not displayed, Superbase creates a multi-line editing box. You can use the normal editing controls to move the insertion point within the field.

If the field is able to include Returns, as specified in the Text Format requester, you can press ENTER inside it to end paragraphs and create blank lines. To exit from such a field, you must press TAB or SHIFT+TAB .

## All Fields or Selected Fields?

The field names that appear may be all the fields or just a few of them. This depends on whether there is a selected fields list active. If you want to enter data into just a few of the fields in the record you should open only those fields using the Field Selection command on the Set menu.

If you do use a selected fields list, Superbase still processes the fields that do not have data entered into them. If a field fails a validation check because you've left it empty, Superbase will tell you.

## Entering New Records in a Form

The steps for starting data entry with a form are almost exactly the same as for a file.

1. Open the form - in the Project menu, point to Open and Form.

   □ The act of opening the form opens all the files needed for the form automatically.

2. Select Record New (AMIGA+N ). Superbase presents a blank form with the insertion point in the first editable field. A blank form might typically appear like this.



   □ If there are any constant fields, their initial values are displayed.

3. Type in the data for the first field, pressing ENTER to finish.

   □ You can exit from data entry at any time by pressing ESCAPE .

4. Repeat step 3 for each field.

   □ After you press ENTER in the last field, Superbase asks whether you want to save the form.

5. Press ENTER to save the form data, or ESCAPE to stop.

   □ If you press ESCAPE to stop, you can go back and make changes and then save the form data with the Save command on the Record menu (AMIGA+S ).

   □ If you press ENTER to save, Superbase asks whether you want to continue.

6. Press ENTER to start again with a blank form, or ESCAPE to stop.

## Saving Form Data

When you save form data with the Record Save command, and fields from more than one file are on the form, Superbase works out which files require updating. If a record from a file has been looked up just so that a field from it may be displayed, but the data for that record has not been changed at all, Superbase does not create a new record for that file. The best way to ensure that looked up data is not changed is to make all the fields that come from that file Read Only on the form itself, even if they are not Read Only in the file definition.

On a multi-file form, all the files are linked through pairs of common reference fields. Only one of the pair actually needs to be on the form. Superbase gives the other one the same value when the data is saved, so that the related records may be retrieved together.

See Chapter 17 Multi-File Applications, for more information on data entry in a multi-file situation.

## Editing Existing Data

In Page or Record View, or when using a form, you can start editing at any time just by clicking in any field. You can use the scroll bars to bring any part of the record or form into view.

You can only edit the current record or form data. So you must first select the data you wish to edit.

You can do this in one of three ways:

• Browsing with a form open.

• Browsing in Record or Page View.

• Browsing in Table View, using the Current Record button to confirm that you have the right record.

• Direct retrieval with the Key Lookup button.

Once you have selected the record you want you can start editing.

1. Click in the field that you want to change, or:

   Select the Current command on the Edit menu (AMIGA+E ).

   □ If Table View is set, Superbase switches to Page View. The insertion point appears in the first field on the screen that is not protected by being defined as Read Only.

2. When you have finished making changes, save the record with the Save command on the Record menu (AMIGA+S ).

### Selecting Another Record

If you have made any changes to the current record or form data, or added any data to a new record, Superbase does not let you perform any other action that involves selecting a record without asking whether you want to go ahead and save first.

# Editing Controls

A set of editing controls allows you to enter and edit data quickly and easily.

### Field Selection with the Mouse

1. Move the pointer over the field you want.

2. Click the mouse.

If you click inside the field, Superbase produces an insertion point at the character on which you click, unless the field is on a form and its data is displayed in a non-system font.

If the field is shown in a non-system font, the insertion point will appear where you click, but since Superbase has to change the field contents into the system font for you to edit it, the insertion point may not be on the actual character you clicked. It is easy to click again, or just use the direction keys to move to the desired place.

In Record View, if you click beyond the end of the field, Superbase moves the insertion point to the end of the existing data; if you click before the start of the field, Superbase moves the insertion point to the start of the field.

### Page and Record View Field Markers

In these two views each field starts one space after the end of the field name zone. In Record View the end of the field that the insertion point is in is marked by a vertical bar. If the field names are shown, all fields in Page View have end of field markers.

### Moving from Field to Field

You may edit any field simply by moving the mouse and clicking.

• In Page View, the UP and DOWN direction keys move the insertion point to the next field above or below the current field (the exact move depends on the position of the insertion point in the field and whether the field names begin in the same column). In Record View, UP and DOWN moves up or down a field respectively.

• ENTER moves the insertion point to the next field. CTRL+ENTER moves the insertion point to the previous field.

- In forms, the data entry order may be defined by the user, so pressing ENTER or DOWN simply moves the insertion point to the next field in the order. Pressing CTRL+ENTER or UP moves the insertion point to the previous field.

## Editing within a Field

| Editing Activity | Key Combination |
|---|---|
| Move insertion point | LEFT and RIGHT direction keys |
| Beginning of field | SHIFT+LEFT |
| End of field | SHIFT+RIGHT |
| Exit from editing | ESCAPE |
| Insert or Typeover | CTRL+V |
| Delete character | DEL |
| Delete character left | BACKSPACE |
| Clear field | CTRL+X |

- Provided that NUM LOCK is switched off (which you can do with the Set System Options command), HOME positions the insertion point at the beginning of the field, END positions it at the end of the field, and INS switches between Insert and Typeover modes.

## Undo Editing

| | |
|---|---|
| Undo field edits | CTRL+U (only possible while the insertion point is still in the field and changes have not been saved) |
| Undo record edits | ESCAPE |
| Reselect the current record | UP direction key (only possible while the changes have not been saved) |

## Entering Dates

For a date field, Superbase uses the format specified for the field in the File Definition as the basis for display. This allows useful short cuts like entering '6 9 89' and having it displayed as 'June 9, 1989'. The only requirement is that you get the Day Month Year order right.

A date in a Validation Formula must conform to the Day Month Year order in the format for the system date as specified with the Date Format command in the Set menu.

## Entering Times

Times must be entered in the format specified in the File Definition.

## Entering Numbers

As with dates, Superbase uses the format specified for a numeric field in the File Definition as the basis for display. Superbase is very tolerant of spaces in your editing. You can enter, for example, '2300.5' and have it neatly redisplayed as '$ 2,300.50'.

## Required Fields

If a field has Required status, you must enter some data into it. If you press ENTER in an empty required field, Superbase displays an error message.

## Read Only Fields

You cannot enter data into a Read Only field. If you attempt to do so, Superbase will move the insertion point on to another field.

## Validated Fields

If you have specified a validation check for a field, any data you enter must pass the check. If it does not, you will see a message like this:
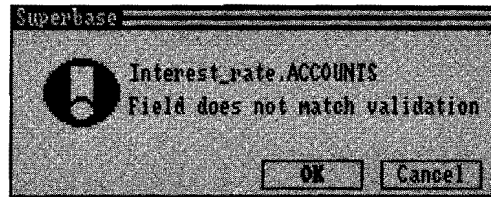


If you already know what the Validation Formula is, then click on the OK button of the message requester. This allows you to continue editing the field.

If you can't remember what the rules for the field are, click on the Cancel button on the requester. Now select Status File from the Utilities menu. This will show you the Validation Formula for the field you are editing. When you understand the rules governing entry into the field, select Current from the Edit menu, click Cancel to retain the incomplete record data, and continue editing. You may want to be able to skip over a field that has a validation check attached. If so, you must include a zero or null option in the Validation formula when you specify it. See Chapter 7 Data Entry Validation.

Validation is only performed when field contents change. If you enter data that fails a validation check and then click Cancel when the error requester appears, the failed data is allowed to remain in the field while you continue working. Simply pressing ENTER in the field does not cause Superbase to register a change. However, Superbase does not allow you to save invalid data, so the error message will re-appear when you select the Record Save command.

### Calculation Fields

Fields that have calculation formulas attached to them will generally be Read Only fields. This means they cannot be edited directly.

If they are not Read Only, you will be able to enter data in the field but it will be overwritten when the calculation is performed – unless it is a self-referencing calculation (see Chapter 8 Derived Values).

When you are entering data into a new record or editing an existing one, Superbase does not display the results of a calculation formula until you:

• Click on the calculated field, or

• Pass through the field, or

• Go to save the record

### Constant Fields

Fields with constant formulas attached to them cannot be edited directly if they are Read Only fields. Otherwise, you can edit these fields in the usual way. Remember that Superbase only works out the result of a constant formula when the record is first created. Thereafter the formula becomes inoperative. Any data you enter will therefore overwrite the result of the formula.

## Creating Duplicate Records

The Record Duplicate command allows you to cut down on the time taken to enter records where some of the data is the same. You make a 'snapshot' of the current record in memory, edit in any changes you want, and then save it in the normal way.

1. Select the record to be duplicated.

2. Select the Record Duplicate command (AMIGA+D ).

   □ Superbase shows a message to confirm that a snapshot record is ready for editing.

3. Make any changes required.

4. Save the record (AMIGA+S ).

Record Duplicate re-initializes any constant formulas. This means you can duplicate a record which contains the constant formula SER without duplicating its serial number. Similarly, if a file uses the constant formula TODAY , creating a new record by duplicating an existing record does not reproduce the original date.

## The Clipboard

The clipboard is a temporary storage area. Because it is common to all standard Workbench applications, you can use it to transfer data between different programs.

The standard commands for transferring data to and from the clipboard are Cut, Copy and Paste. If you cut or copy data using another program, you can then switch to Superbase and paste the same data into a Superbase file. And the process also works the other way round: other programs can make use of Superbase data which has been placed on the clipboard.

Within Superbase, the clipboard also serves as a trash can allowing you to remove data permanently.

It is important to understand that the clipboard can only hold one set of data at a time. If you cut or copy a record or a field, it will overwrite any data that is already stored in the clipboard. Pasting data, on the other hand, leaves the clipboard intact. You can paste the same set of data as many times as you wish.

With a multi-file form, only data from the current file is placed on the clipboard. You may have to select each file in turn if you wish to remove all the data that appears on a form.

**Note**    Some applications can store both text and graphics on the clipboard; Superbase only stores text. Note also that when you Cut or Copy records to the clipboard, fields are stored on one line separated by tab characters (eight spaces).

## Field Level Cut, Copy and Paste

At field level, the Cut, Copy and Paste commands operate only on the whole field.

A copy of the data in the selected field may be placed on the clipboard by Cut (AMIGA+X ) and Copy (AMIGA+C ), and removed from the field by Cut. Paste (AMIGA+V ) reads data into the field from the clipboard.

You may use this facility when you are creating new records which contain the same data in one of their fields. First copy the field data to the clipboard, then paste it into the same field in each of the new records.

Another use is in conjunction with the Superbase Text Editor. Data can be written to the clipboard from within the Text Editor, and then pasted into individual fields in database records.

# Cut

From time to time you may find that a record is no longer needed. Provided the insertion point is not active, this menu command removes the current record and stores it on the clipboard.

1.  Make sure that the insertion point is not present (press ESCAPE if necessary).

2.  Select the record you want to remove.

3.  Select Cut from the Edit Menu or press AMIGA+X .

If you discover that you have made a mistake and have removed the wrong record, you can restore it using the Paste command – provided you have not used Cut or Copy with another record in the intervening period.

# Copy

Use this command if you want to place an exact copy of the current record on the clipboard.

1.  Make sure that the insertion point is not present (press ESCAPE if necessary).

2.  Select the record you want to copy.

3.  Select Copy from the Edit Menu or press AMIGA+C .

Before copying a record, make sure that the insertion point is not active. Then select Copy from the menu or press AMIGA+C .

# Paste

Paste reads the contents of the clipboard into the database file. If there is an insertion point in a field, the data is pasted into that field.

When there is no insertion point in any field, Paste takes the data on the clipboard and inserts it in the current file. If you have previously cut or copied a record from another file, selecting Paste from the menu (or pressing AMIGA+V ) will store it as a new record in the current file.

•   You do not need to save the record after using Paste.

## Pasting Multiple Records

It is possible for the clipboard to contain data for more than one record; this could occur, for example, if you cut or copied rows from a Workbench spreadsheet. In this case, Paste will create multiple records in the current file. Superbase shows a requester to indicate how many records have been pasted

into the file.

- The clipboard data must be in the correct format for the current file. For example, dates and numbers must be valid.

See also Chapter 23 Editing Text.

# 7 DATA ENTRY VALIDATION

Superbase allows you to attach a validation formula to a field. The formula may be thought of as a rule that regulates what can be entered into the field. This allows the application designer to simplify the data entry process for the user by automatic checking of field values against range limits, patterns, and lists of alternatives held within the formula. Cross-file lookup may also be used to check entered data against data already on file. These basic rules are useful in a single file situation, and crucial in a multi-file situation such as data entry into a multi-file form.

Validation formulas may also be added to fields on forms. In this situation, the validation acts as a secondary rule governing data entry for the particular form. As you can design many different forms for every file, you may complement the file level validations which comprise the fundamental rules for data entry with extra rules at the form level for different application areas or users.

Validation formulas can be created either during the file definition process or at a later stage when you are modifying a file. See Chapter 3 Defining a New File and Chapter 5 Modifying Existing Files. The procedure for constructing a validation formula is described further on in this chapter.

## Types of Validation

There are four main ways of validating data as it is entered in a record: using a range, a list, a pattern, or a cross-file LOOKUP .

### Ranges

This type of validation is usually applied to numeric or date fields. By imposing an upper or lower limit, or both, it restricts data to a range of values.

Examples are:

    amount.Clients >= 100 AND amount.Clients <= 200

This requires the entry (into the **amount** field) to be between 100 and 200 inclusive.

    date.Clients > "6/1/89"

The date entered must be after June 1, 1989. Note that the interpretation of the Day/Month/Year order of the date will depend on the system.

## Lists

By using the logical operator OR in a validation formula, you can specify a list of acceptable values. Superbase will only validate data as it is entered if it matches one of the items in the list.

For example:

> firstname.Clients LIKE "Tom" OR firstname.Clients LIKE "Jerry"

would restrict the **firstname** field to the values 'Tom' or 'Jerry'. Note the use of LIKE instead of the '=' operator. LIKE is case insensitive and allows you to enter text in either capital letters or lower case, or a mixture of both.

> Country.Clients = "USA" OR Country.Clients = "France" OR
> Country.Clients = "England" OR Country.Clients = "Italy"

Here the formula restricts data in the **Country** field to four items. Unlike the previous example, it also requires the user to enter the country in the correct format – with the first letter capitalized.

## Patterns

By using LIKE, you may define a formula which limits text entry to a specified pattern. Here is a typical example:

> Lastname LIKE "[A-F]*"

This formula accepts an entry of any length, provided it begins with a letter from 'A' to 'F' (either case). More details on pattern matching with LIKE are in Chapter 10 Retrieving Data Interactively and Chapter 32 Learning to Use Superbase.

## Cross-File LOOKUP

The LOOKUP function can be used in a validation formula to check data entry against the contents of a field in another file.

The syntax for LOOKUP is:

> LOOKUP(field1.file1,field2.file2)

**field1.file1** represents the name of the field to which the validation formula is attached, i.e., the current field. **field2.file2** represents the name of the field in another file.

When you enter data in a field which has a LOOKUP validation, Superbase checks whether it matches the data in any of the records in the other file. If it doesn't, Superbase rejects your entry.

See Chapter 18 Linking Files for more information on the LOOKUP function.

# Constructing a Validation Formula

You can create a validation formula for a field when you first define the file or, at a later stage, when you are modifying the file definition. Either way, you construct a validation formula using the File Definition requester.

A validation formula can only be attached to a field after it has been Added to the Field Name box (using the Add button). The reason for this is that a validation formula contains the name of the field it validates. If the field has not yet been defined, Superbase will not recognize the field name when you include it in a formula.

The correct procedure, therefore, is to Add the field first and then create the formula. In effect, this means you have to define the field first, and then edit its definition.

For example, if you wanted to create a date field with a formula attached to it, you would need to carry out these steps first:

1. Define the field as a date field and set the date format.

2. Click Add – the field name will then be added to the File Definition panel.

3. Click the field in the File Definition panel to select it for editing.

Once you have completed this preliminary task, you can proceed to the next stage where you construct the formula and enter it in the Validation Formula requester.

1. Click the Validation check box. This action will present you with the Validation Formula requester:

2. Build up the validation formula using the field list box, the operator buttons and value box. You click an operator or a field name to place it in the Text box; and you can add other elements in the formula – numbers, text, dates, etc. by typing them into the Value box and pressing ENTER.

3. Alternatively, you can type the formula directly into the Text box. You will need to do this if the formula contains functions such as LOOKUP .

4. When you've finished the formula, click OK. If you want to clear the formula from the requester and start again, click Clear.

## Validation Help Messages

A validation formula sets limits to the data that can be entered in a field. If you attempt to enter invalid data, Superbase will reject your data entry. At the same time, it displays a validation error message:
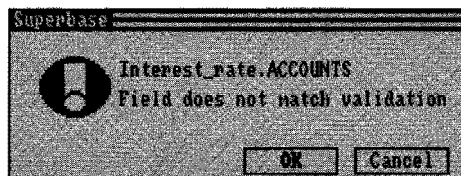
For this message, we've taken the validation formula which is assigned to an **Interest Rate** field:

> Interest Rate >= 0 AND Interest Rate <= 15

This formula restricts the user to values between 0 and 15.

The Validation Help feature allows you to make the message even more explicit. Instead of displaying the field name in the first line of the message, you can include a specific error message which tells the user exactly why the data is invalid.

Validation Help messages are defined as part of the formula. The procedure is as follows:

1. Construct the validation formula, as explained earlier.

2. Click in the text box and type ELSE at the end of the formula.

3. Then type the help message, enclosing it in quotation marks.

4. Click OK.

If you wanted to modify the formula for the **Interest Rate** field in the example above, you would enter:

> Interest Rate = 0 AND Interest Rate < 15 ELSE "Interest
> Rate too high or less than zero"

Now, the validation help message will look like this:
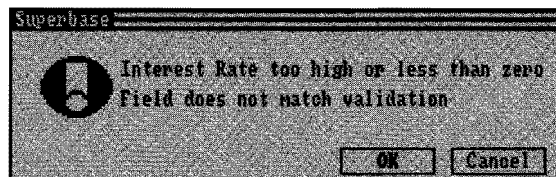


## Further Information

Here are some useful facts about validation formulas:

* The maximum length of a validation formula is 255 characters.

* You can place field names on either side of an operator:

  field 1 > field 2

- Superbase stops you making simple mistakes. It does this by flashing the screen if a validation fails.

- You can use parentheses to set priority levels for your formula:

  amount.Clients = 10 OR ( amount.Clients > 20 AND
  amount.Clients < 30 )

- To allow the user to create a blank Numeric or Date field, you must include a 'zero option' in the formula:

  amount.Clients > 10 OR amount.Clients = 0

  The same applies to text fields, except you type "" into the Value Box instead of 0.

- If you click in the Text box, you can edit the formula directly. This is essential for making changes to an existing formula or for adding functions which are not available from the selections in the requester. See Chapter 9 Using Functions.

# 8 DERIVED VALUES

In addition to the data you enter directly into records, you can use calculation formulas to derive new values from existing data. This chapter describes the use of calculations to generate values which are then stored in fields. Derived values can also play a useful role in queries and updates. See Chapter 14 Defining and Using Queries and Chapter 19 Updating.

Calculation formulas may also appear on forms, providing additional processing power beyond the file level. Form calculations can generate running totals, transfer data from one file to another in a multi-file form, or serve as programmable variables. Forms may also include command lines, providing an interface to DML programs. See Volume 2, Form Designer User Guide.

## Field Calculation Formulas

A calculation formula works out and stores a result that depends on other fields and values. Unlike a validation formula, which checks data entered by the user, a calculation formula generates the data itself. This means that you cannot edit a calculated field directly. To be more precise, if the field is not Read Only, you can attempt to edit its contents, but the formula will then overwrite your entry with the calculated result.

Calculation formulas can be attached to any type of field. Calculated fields may also be validated, so that the result is first calculated and then checked.

Here are some examples of typical calculation formulas:

```
quantity.Invline * price.Stocks
subtotal_1.Orders + subtotal_2.Orders + subtotal_3.Orders
amount.Stocks * ( 1 + increase.Stocks / 100 )
```

Formulas are usually arithmetical and attached to numeric fields, and as such rarely make use of relational operators like 'greater than' or 'less than'.

There are restrictions on the combinations of types of field in a formula expression. Superbase will let you know if you break a rule, by trying to multiply two text fields, for example. Remember, though, that you can treat date and time fields as numbers.

You can create formulas for text fields by typing directly in the Formula box:

```
LEFT$(Lastname.Clients,5) + "/" + STR$ (SER("Clients"),"00000")
```

Here the formula makes use of the Superbase functions, LEFT$, STR$ and SER (see Chapter 9 Use of Functions). It could produce a result like 'johns/00042', which could in turn be the basis of an index.

Date and time formulas are also valid. Typically, they would be used with the reserved words TODAY and NOW, to show the date and time when a record was created or last edited.

All the examples given here relate to single file applications. Chapter 18 Linking Files explains how formulas can also be used to extract data from other files for insertion in the current file.

## Constructing a Calculation Formula

You do this in the same way as you create a validation formula (see Chapter 7 Data Entry Validation), with one important difference: you do not need to include the name of the field you are working with in the formula itself. If you wanted to create a formula which added two fields together and placed the result in a field called **total**, you would enter:

    amount1.Orders  +  amount2.Orders

Unless you are using the ternary operator (explained in a later section in this chapter), a calculation formula should not contain the equals sign; so it would be a mistake to enter:

    total.Orders  =  amount1.Orders  +  amount2.Orders

When you create a formula, the field you are attaching it to is shown at the top of the calculation requesters; and the result of the formula is automatically assigned to that field. This is why it is not necessary to include the field name in the formula itself.

It is possible, however, to construct a formula which includes the name of the field it is attached to. We refer to this as a self-referencing formula. For reasons which are explained further on in this chapter, you should exercise caution before creating formulas of this type.

## Using Dates and Times in Formulas

Superbase stores dates by representing them as julian date numbers; a date's julian date number is the number of days between the date and the year A.D. 0. For example, the date, 26 July 1986, has the julian date number 725202; and the same date a year later has the value 725202 + 365.

This method of storing dates makes it easy to incorporate date fields in formulas with numeric values. For example, to generate the number of days between two dates, or to set a date in the future:

    date1.Orders - date2.Orders
    date.Orders + 90

Time fields can also be treated as numeric fields. Superbase stores times as thousandths of a second within a 24 hour period.

## Using the System Variables

'System variables' is the term we use to describe the Superbase keywords, TODAY and NOW. Your computer system keeps track of the date and time; these words provide a means of displaying the date and time, and inserting them in a field.

If your system has a battery-backed real-time clock, TODAY and NOW will show the current date and time; that is, the correct date and the correct time. Otherwise you can set the date and time using the Date Format command on the Set menu, at the start of any session with Superbase. For the rest of the session, the system will then update these variables constantly, so that they show the current date and time.

By using TODAY and NOW in calculation formulas, you can ensure that the current date and time are inserted in a record whenever it is edited or updated. Most people do not need to know the exact time when a record was updated, so NOW is less likely to be used in a formula. But it is often useful to 'date stamp' a record using the TODAY variable. To do this, first define a date field, then create a formula consisting simply of the word TODAY.

Calculation formulas are not suitable if you want to show when a record was first created. For this, you need a constant formula (see the section **Constants** further on in this chapter).

**Note** Note that NOW and TODAY can also figure in validation formulas. For example, you could use TODAY to invalidate checks that are out of date:

Checkdate > (TODAY - 365)

**Caution** Do not use TODAY and NOW in a calculation which is attached to an indexed virtual field.

## When is a Calculation Performed?

A record may contain just one or a number of calculated fields. When you save a record, Superbase carries out all the calculations in the record and stores the results in the fields which the calculation formulas refer to. This applies both to new records and to edited records.

Most, if not all, calculated fields will be Read Only. With these fields, calculations may also be performed while you are entering data in the record or editing it. This happens if you click in the field or if you pass through the field on the way to another field further down the field list. The result will be that the calculation is performed at least twice: every time you click in it and then again when you save the record.

You can see this process in action by defining a time field with the calculation formula NOW attached to it. Provided you have set the time format so that

seconds and milliseconds are shown, the time field will be updated every time you click in it.

Usually, repeating a calculation gives the same result, and it doesn't matter how many times the calculation is carried out. Likewise, it doesn't usually matter in what order you enter data in fields, or edit fields. The only time these considerations may be important is when a record contains linked calculations, as explained in the next section.

## Linked Calculations

It is sometimes important to consider the order in which calculations are carried out. If they are carried out in the wrong order, you may get the wrong result. The problem only arises with linked calculations, where the result of one calculation depends on the result of another.

Normally, you do not need to take the order into account. When you save a record, Superbase calculates fields in the order in which they occur in the field list; that is, it works out the result for a calculated field at the top of the record (in Record View) first and then passes to the next calculated field below.

Incorrect results are only produced when the order in which the records should be calculated does not match the field list order. The following example shows how this might occur.

**Example**

This example presents two sets of calculations. In the first set the order in which the calculations should be performed matches the field list order and gives the correct results:

| Field | Calculation | Value |
|-------|-------------|-------|
| Unit Cost | | $2.40 |
| Quantity | | 10 |
| Total Cost | Unit Cost * Quantity | $24.00 |
| Discount Rate | | 20% |
| Discount | (Total Cost * Discount Rate)/100 | $4.80 |
| Total Amount | Total Cost - Discount | $19.20 |

Here, the calculation formulas for **Discount**, and **Total Amount** are linked to the formula which is attached to **Total Cost**. These formulas generate the data values for their associated fields. The values in the other fields are fixed values entered by the user.

What would happen if you now decided to modify the file definition in order to introduce two more fields, **Tax Rate** and **Tax Amount**? This step would be necessary if you wanted to include the local tax rate in your calculations. You

would be able to modify the existing formulas in the file definition, but the two new fields would have to be added at the end of the field list. And the formula attached to **Tax Amount** would be calculated in the wrong order.

After all the fields have been calculated once, the results would be as follows:

| Field | Calculation | Value |
|---|---|---|
| Unit Cost | | $2.40 |
| Quantity | | 10 |
| Total Cost | Unit Cost * Quantity | $24.00 |
| Discount Rate | | 20% |
| Discount | (Total Cost * Discount Rate)/100 | $4.80 |
| Total Amount | Total Cost - Discount + Tax Amount | $19.20 |
| Tax Rate | | 15% |
| Tax Amount | (Total Cost - Discount) * Tax Rate/100 | $2.88 |

You will notice that the result for **Total Amount** is incorrect, despite the fact that **Tax Amount** contains the correct value. The reason for this is that Superbase calculates the value for **Total Amount** before it calculates the value for **Tax Amount**. When the value for **Total Amount** is calculated, the value of **Tax Amount** is 0.00.

**Calculation Count**

Superbase supplies a straightforward solution to this problem. It is based on the fact that if you perform the calculations a second time, you will arrive at the correct result. You can do this by setting the Calculation Count option in the Set System Options requester.

This feature allows you to specify the number of times a calculation (or a chain of calculations) is performed. With a more complicated set of linked calculations you may need to perform them three or four (or more) times before arriving at the correct result. In each case, you would set the Calculation Count to the number required. For further details, see Chapter 31 Customizing Your System.

Another solution would be to create a new file in which the fields were in the correct order. To do this, you would enter the fields in the correct order in Query Fields command line, and then use the Query to File option to create a new file from the existing file. See Chapter 16 Reorganizing the Database for more details.

## Self-referencing Formulas

A calculation formula should not normally contain the name of the field it is attached to. But if you edit a calculation formula after it has been added to the file definition, it is possible to insert the field name in the formula, or to create a new formula which includes the name. In this way, you could create a self-referencing formula for the field **total**, such as:

> total + amount

When the formula is calculated the value in the field **amount** will be added to the value in **total** and then stored in **total**.

Although this may appear to be a useful feature, this type of self- referencing formula is not recommended. The following example should make this clear.

Suppose the initial values for the two fields **total** and **amount** were:

> amount: 25
> total: 50

After the formula has been calculated once, the new values will be:

> amount: 25
> total: 75

This, of course, is the correct result. But if the formula is calculated again, the values will be:

> amount: 25
> total: 100

As you see, if the formula is calculated more than once, it gives the wrong result. Because it is difficult to ensure that a formula is only calculated once, self-referencing formulas of this type will invariably give the wrong result and should therefore be avoided.

However, there are some self-referencing formulas that will always produce the desired result, no matter how many times they are calculated. Formulas which include the ternary operator (see the next section) provide one example. Another example would be a formula like this:

> LTRIM$(Firstname)

LTRIM$ strips leading spaces from the text specified in the brackets. You might want to attach this formula to the **Firstname** field, in order to eliminate any leading spaces that the user might type in accidentally.

Notice that this also provides an example of a calculated field which is not Read Only. When you define a calculation formula, Superbase automatically makes the field Read Only. If the field is intended to be used for data entry, you will need to turn the Read Only attribute off after defining the formula.

## Checklist

- When you attach a calculation formula to a field, it is not necessary to include the field name in the formula.

- Superbase can stop you making simple mistakes, but you can still produce meaningless formulas if you try hard enough, or if you edit the formula directly in the Text box.

- If you need to construct a very complex formula, it may be easier to set up some intermediate calculation formulas than to try to achieve the desired result in a single very large formula.

- The maximum length of a calculation formula is 255 characters.

- You may use a formula for cross-file calculation, where it extracts data from another file.

- The number of decimal places in a calculation result may be set with the directly entered FIX function, e.g. FIX (amount/3,2).

# Constants

A constant formula works in the same way as a calculation formula. It generates a result from other fields, functions, and values, and places it in the field the formula is attached to. The difference is that the result is only generated once when the record is first created. After that the formula becomes defunct and no longer operates.

If you make a constant field Read Only, the field will retain its initial data no matter what changes you make elsewhere in the record. If the constant field is not Read Only, the initial data it contains can be edited or overwritten.

Typical applications for constant formulas are:

- To show the date when a record is created.

- To set the record's serial number using the SER function.

- To set some other kind of reference field such as an account number or a customer number.

# Generating Serial Numbers

You can use the SER function to assign a serial number to each record in a file. To do this:

1.  Define a numeric field which will hold the serial number.

2.  In the File Definition requester, click on the field name to place it the Field box.

3.  Click on the Constant button.

4.  Enter: SER("filename")as the constant formula for the field. For example, to assign a serial number to the records in a file called **Orders**, first define a constant field for the number, then attach the formula: SER("Orders")

When you create the first record, it will be given the value 1. This value will then be incremented by one for each record you add to the file.

# Using the Ternary Operator

The ternary operator is one of Superbase's more unusual features. It may take you a while to grasp the idea behind it, but it is certainly worth the effort. Put simply, the ternary operator allows you to create calculation formulas which can choose between two (or more) alternative results. Formulas like this have a decision-making ability built into them.

Suppose you wanted to create a file which stored the results of a school examination. One way of doing this would be to use one record for each student. The record would include a **marks** field, showing the marks gained in the examination as a percentage figure. It could also include a text field called **grade** which showed at a glance whether a student has passed or failed. By using the ternary operator we can define a formula which sets **grade** to either 'Pass' or 'Fail' according to the student's percentage mark:

> (mark.Results >= 75) ? "Pass": "Fail"

The effect of this formula could be translated as:

> If the number in the field **mark** is greater than or equal to 75,

store 'Pass' in **grade**, otherwise store 'Fail' in **grade**.

The general form (or syntax) of the ternary operator is this:

> *condition* ? *value1* : *value2*

The words in italics represent the specific values or conditions that are entered in a formula. In the example above, 'Pass' is *value1*, and 'Fail' is *value2*.

The *condition* argument must be an expression that is either true or false, such as:

    date.Clients > "18 July 1990"

or

    initial.Clients = "B"

The parentheses around the condition are optional, but it is a good idea to include them. By separating the condition from the other parts of the ternary operator, the parentheses make it easier to understand how the formula works.

If you are familiar with programming languages, you will recognize the ternary operator as a more concise way of expressing the IF THEN ELSE statement. We could translate the example above to read:

    IF  mark  >= 75
    THEN  grade =  "Pass"
    ELSE  grade =  "Fail"

## Extended Ternary Operations

Instead of entering a value as the second alternative in a ternary operation, you can enter another ternary operator. In this way, you can create a formula which chooses between three (or more) alternatives:

    (mark.Results >= 75) ? "A" : (mark.Results >= 60 ) ? "B" : "C"

Here, our example formula sets **grade** to 'A', 'B' or 'C'. Marks greater than or equal to 75% are graded A, marks between 60% and 74% are graded B, any other mark is graded C.

This formula can be modified again to provide further alternatives:

    (mark.Results >= 75) ? "A" : (mark.Results > = 65) ? "B" :
    (mark.Results >= 55) ? "C" : (mark.Results >= 45) ? "D": "E"

The number of alternatives you can build into a formula is limited by the maximum number of characters. A formula must not exceed 255 characters, including spaces.

## Self-referencing Ternary Operations

The ternary operator proves one of the exceptions to the rule we stated earlier, which advised against creating self-referencing formulas. Normally, the drawback with this type of formula is that it may give a different result every time it is calculated. Using the ternary operator, you can avoid the problem by placing the field name (the name of the field to which the formula is attached) at the end of the formula. In other words, the last value in the formula should be the field itself. There is no other way of making a self-referencing formula totally foolproof.

Take, for example, the formula:

    code.Address = "a" ? "London": code.Address = "b" ? "New
    York" :"Other"

The first time you enter the letter 'a' in the field **code.Address** the formula
will give the result 'London'. But if the formula is calculated again, it will give
the result 'Other'. The reason for this is that **code.Address** will contain the
word 'London', not the letter 'a'. This value does not satisfy either of the two
conditions so it is replaced with 'Other'. The correct way of insuring against a
false result like this would be:

    code.Address = "a" ? "London": code.Address = "b" ? "New
    York": code.Address

Now when the formula is calculated a second time, it leaves the result as it is.
None of the conditions are satisfied, so the field is assigned the value in the
last alternative. Since the last alternative refers to the data already in the field,
the initial value remains unchanged.

## Further Information

You will find a list of the Superbase functions in Appendix B, together with a
brief description of each. Each function is described in detail in Volume 2,
DML Reference Guide. See also Chapter 9 Using Functions.

# 9 USING FUNCTIONS

Superbase provides many powerful functions for use in different types of formula. These functions, which are part of the Superbase Database Management Language, itself a superset of BASIC, give you access to a deeper layer of the Superbase system. However, you do not have to be a programmer to use functions effectively. In this section we look at some of the ways in which these functions can be used in practice. A complete list of functions together with the syntax for each one may be found in Appendix B. Each function is described in detail in Volume 2, DML Reference Guide.

## Where To Use Functions

Functions may be used in formulas, in filter command lines, in queries and in updates.

Validation, Calculation and Constant formulas are part of File Definitions. Filter command lines are used with the browsing controls, and all the Process menu options except for Reorganize. In Queries, functions are allowed in both the Fields and Filter requesters. In Updates, formulas are allowed in both the Filter and the Update Definition requesters.

## Types of Function

Functions may be broadly categorized as string, date, time, financial, report, and mathematical or numeric. Within some of these there are sub-categories.

**Numeric functions** are transcendental (COS, SIN, TAN, etc.); BASIC (VAL, ASC, LEN, INSTR); mathematical (INT, ABS, SQR, etc.); and system (DISKSPACE, SER, EXISTS, LOOKUP, etc.).

**String functions** are mostly the same as BASIC (STR$, LEFT$, MID$, TRIM$, CHR$, etc.).

**Date functions** (DAY$, MONTH, YEAR, etc.) manipulate date fields and julian values, and **time functions** (TIME$, SECS, THOUSECS, etc.) manipulate time fields and values.

**Financial functions** (FN SLN, FN FV, FN NPER, FN PMT, FN PV, FN RATE) cover most requirements for interest, amortization, and depreciation calculations.

**Report functions** (SUM, MIN, COUNT, MAX, MEAN, VAR, SD) may only be used in forms, reports, and DML. SUM, COUNT and MEAN are also valid keywords in Queries.

Each function may only be used with appropriate values and fields. For example, you may not use a formula such as LEFT$(123); this would produce the error message 'Data types don't match.'

## How to Use Functions

Wherever functions are used, they must be typed into formulas and expressions, following the correct syntax as described in Appendix B. See Chapter 7 Data Entry Validation and Chapter 8 Derived Values for details of how to create formulas and expressions.

When typing in functions, be sure to enter parentheses in pairs: if you have three '(' characters you must also have three ')' characters.

**Validation formulas** contain rules to govern the entry of data into fields. A validation formula must yield a true or false result. For example, you might wish to check that a date entered does not fall on a weekend. For this, the formula could be:

> DAY$(datefield) <> "Saturday" AND DAY$(datefield) <> "Sunday"

If the day of the week generated by DAY$ is neither 'Saturday' nor 'Sunday', then the formula will yield a 'true' result and the data entered into datefield will have passed the validation.

Alternative ways of expressing the same thing are:

> NOT (DAY$(datefield) LIKE "Saturday" OR DAY$(datefield) LIKE "Sunday")

or:

> INSTR ("SaturdaySunday",DAY$(datefield))=0

The first of these is case insensitive. Notice the use in these three formulas of the operators AND, OR, NOT, and LIKE.

The third example is more advanced. It demonstrates the use of a true/false test to perform a validation. Any expression that uses one of the Relational Operators (= = > < <> LIKE CONTAINS) yields a true or false result. If the result is true its numeric value is -1; if the result is false the value is 0. So in the example above, if the day of the week of the date typed into the datefield *does* occur in the string expression 'SaturdaySunday', the result of the INSTR function will be true, -1, and the validation rule, which requires the result to be 0, will have been broken.

LOOKUP is a very important function used in validation formulas for linking between files. See Chapter 18 Linking Files.

**Calculation formulas** derive results from expressions. A calculation formula can use functions to derive more specialized results than can be produced from ordinary arithmetic expressions. The following examples show how this can be done.

> INT(SQR(numfield))

Generates the integer portion of the square root of **numfield**.

ATN(numfield/SQR(numfield*numfield-1))+
SGN(SGN(numfield)-1)*PI/2

Generates the inverse secant of **numfield**.

YEAR(datefield)+1

Generates the numeric value of the year after the year of **datefield**.

RIGHT$(STR$(RND(-1)+1000000),6)

Generates a six character string of a random number with leading zeros.

DATE$(DAYS(datefield),"0d-mm-yy")

This formula applies to **datefield** itself. It generates a date with a leading zero for single digit dates. **datefield** must be Text type, and the calculation must not be Read Only. Entries into datefield will be validated as legal dates, and reformatted according to the date string.

LTRIM$(fieldname)

The LTRIM$ function eliminates leading spaces from a text field. The calculation must not be Read Only.

**Constant formulas** are used to set initial values in data entry. Functions are often used to generate key field data automatically. For example, you could generate a unique four character key from the record serial number like this:

RIGHT$(STR$(SER("filename")+10000,"99999"),4)

**Filters** are used when browsing or performing Process menu operations to define a set of records for viewing or processing. The filter is like a validation formula in that it must yield a true or false result for every record read from the file. Functions are used in filters in exactly the same way as in validation formulas.

**Derived expressions** extend the power of Query and Update by allowing you to generate results on the fly as data is output or records are updated. Derived expressions are like calculation formulas in that they yield results. Expressions can range from a simple multiplication of two fields to a complex combination of mathematical functions. The calculation formula examples above show how functions operate in derived expressions.

## Special Functions

Certain functions have special roles in the Superbase system. These are detailed elsewhere in the manual, but are grouped here for convenience.

**DDE$**

DDE$ is included for compatibility (programs written for other computers may use this function). On the Amiga this function does nothing.

## FIX

FIX is used with numbers such as currency amounts to eliminate rounding errors caused by differences between display format and true numeric precision. It sets a real limit to the number of decimal places computed.

## LOOKUP

LOOKUP is used to establish links between files for relational data entry. See Chapter 18 Linking Files.

## PAD$

PAD$ is important for multi-field indexing. It is used in the calculation formula that generates the key to regulate the length of an element. See Chapter 4 Creating Indexes.

## SER

SER increments a serial number for the current file every time you create a new record. See Chapter 8 Derived Values.

# 10 RETRIEVING DATA INTERACTIVELY

Superbase provides a number of controls for browsing through files and selecting records. These operate both for single files shown in one of the three default views and for forms which refer to more than one file. Most of the explanations in this chapter apply equally to browsing through both files and forms.

## Opening Files

Select Open and File from the Project menu. Superbase displays this requester:
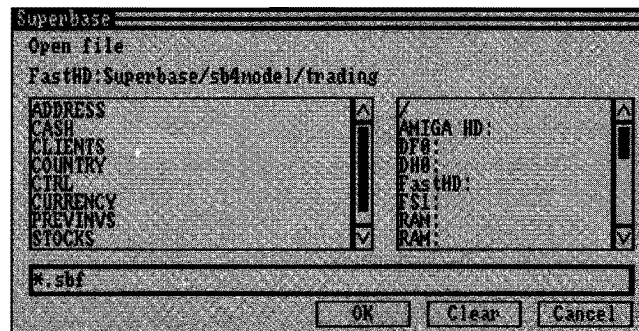


In the box on the left, Superbase presents you with a list of all the files in the current directory. The list box on the right allows you to change to another directory. To open a file:

1.  Click a file name and when it appears in the text box, click OK.

or:

2.  Double-click the file name.

If you want to alter the name in the text box, you can:

*   Click another name to replace it.
*   Click Clear to empty the box.
*   Click in the text box and edit the contents. You might need to do this if your file is in another directory and you need to insert the pathname.

If you wish to open a dBase file, click the dBase checkbox. See Chapter 11 Working with dBase Files.

## Opening the Current File

When you select a file name and click OK, Superbase finds the File Definition for that file and returns you to the Work Area.

If the file is being opened for the first time in the session, Superbase automatically selects the first index for the file, determined by the position of the index field in the File Definition. The record selected for display is then the first record according to this index.

If you have already opened the file during the session, Superbase returns to whichever index and record were current when the file was last used.

The file you see on the Work Area is known as the current file. You can always tell which is the current file by looking at the title bar of the main display window, which names both the file and its current index:

    Superbase - FASTHD:SUPERBASE/KTEST1 indexed on firstname

There is only one current file at a time, although you may have many files open.

## Current File and Open Files

You can return to the Open File requester and open as many files as you want. Each time you select a file, it becomes the current file. However, all the files you select in one session remain open and available to other Superbase functions, unless you specifically close them with the Close command.

Superbase functions that refer to the list of open files are:

**Project Remove File**
**Process Update**
**Process Query**
**Utilities Status System**

## Changing the Current File

If a number of files are open at the same time, pressing the '+' key on the numeric keypad cycles through them, making each in turn the current file. The other way of changing the current file is to use the Open command: if you have already opened a file, selecting it again from the Project Open File requester makes it the current file.

## Closing a File

You may want to close a file to release some memory back to the system, or perhaps because you have already have a file open when you wish to open another file of the same name in a different directory.

1.  Make the file you wish to close the current file.

2.  Select the Project Close File command.

**Note**     If you want to close all the open files at once, select the Project Close All command.

# Opening Forms

Select Open and Form from the Project menu. Superbase displays a standard file selector requester that lists any Superbase form files, with the SBV extension, in the current directory.

To open a form, proceed exactly as for a file.

When a form file is opened, it opens any files it needs that are not already open. For a multi-file form, the initial display shows the first record in the master file and any related records.

The master file, as defined in the Form Designer SB File Link command, becomes the current file, indexed on its default index.

You may return to the Open File requester at any time to choose another file as the current file.

## Report Forms

The Form Designer generates a SBV file for both a screen form and a report form. Report form SBV files are also listed in the Open Form requester. If you open a report form, Superbase presents you with the option to execute the associated report program.

# Open Index

The index that you select determines the order in which the records in the current file are presented. If, for example, you have a file of clients for which Superbase maintains indexes on the **lastname** field and the **city** field, you can choose to view or print the records in the alphabetical order of either **lastname** or **city**.

Superbase allows only one current index at a time, as indicated on the title bar of the Work Area. There are a number of important points to note about the restriction to one index:

*   It does not restrict your ability to specify multiple values when searching for a group of records in the file using a filter.
*   It does not limit your ability to produce reports with many levels of sorting and subtotaling.
*   When you are modifying a record or adding new data, all the indexes in a file are updated – whether they are open or not.

- You may create an index which applies to more than one field at a time. See Composite Indexes, Chapter 4 Creating Indexes.

When you select the Index Open command, Superbase displays a requester with a list of all the indexes for the current file.
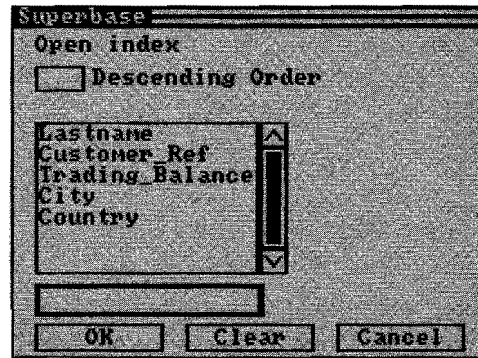


Choosing an index from the requester is easy:

1. Click the name of the index you want. When the name appears in the text box, click OK.

or:

2. Double-click the index name.

If you make the wrong selection, change it by clicking on another index, clicking on Clear, or typing directly into the text box.

When you click OK, Superbase opens the required index and returns to the Work Area. The name of the open index appears on the title bar:

Superbase - FASTHD:SUPERBASE/KTEST1 indexed on lastname

All browsing operations and Project Print File now follow the order of the index. The Key Lookup button allows you to type in a single index entry in order to retrieve an individual record directly.

## Changing the Index

There are two ways of switching from one index to another. First, you can use the Index Open command and select the required index as described above. Second, you can press the '-' key on the numeric keypad. This cycles through all the indexes in a file, making each one in turn the current index.

When you change the index, Superbase will not change the current record but all browsing operations will follow the order of the new index.

If you need a new index on a field that doesn't have one, you must use the New Index command to build the index. See Chapter 4 Creating Indexes.

Each index points to one record in its file. This means you can swap to another index, browse through the file in the order of that index, and then return to the first index and view the record it was originally pointing at.

## Reversing Index Order

The order in which indexed fields are shown is normally ascending. For text fields this means sorting from A through Z. Date fields are sorted from earlier to later, and numeric from smaller to larger.

You may prefer to reverse the order used by a particular index during a session, for example if you wanted to step through records using a diminishing numeric or amount value.

1. Select the File Index Open requester.

2. Choose the index to be reversed.

3. Click the Descending order check box.

4. Click OK.

The index appears with a (D) after it in the window title bar. The browsing controls will now operate in reverse, so that, for example, Fast Forward with a date index will go from later to earlier.

# Browsing Controls

The browsing controls at the bottom of the screen provide a variety of ways of selecting records according to the current index. Each button has a special purpose, and together they fall into three groups. On the left are the Pause and Stop buttons, followed by seven buttons for browsing through the file; for those who prefer to use the keyboard rather than the mouse, each of these buttons has a keyboard equivalent. On the right are three special purpose buttons: the Key Lookup, Filter, and Camera buttons. The first two are explained later in this chapter; for more information on the Camera button, see Chapter 28 Integrating Text and Image Files.

With a single file in Record, Page, or Table View, you select one record at a time.

You are not confined to one index. You could browse through the **Authors** file equally well by **Lastname** or by **Date of Birth**, provided you had indexes on both fields.

## Forms

Multi-file forms are more complex. When you create a multi-file form in the Form Designer, you define a link structure that informs Superbase how to retrieve related records during browsing. The link structure includes the

concept of a master file. This is simply the file that Superbase uses to set the default browsing order, and its name is shown in the window title bar as the current file.

When you select the next record in the master file, Superbase retrieves all the related records from other files. For example, if your form showed details of authors and their books, and **Authors** was the master file, selecting the next author would also display records from the **Books** file that were linked to that author via a common reference field such as **Author_code**. See Chapter 17 Multi-File Applications for more discussion of these concepts.

The master file of a form need not be the current file. If you make one of the other files on a form the current file (by pressing the '+' key) then browsing will use that file as the temporary master file. So you could step through the **Books** file retrieving details from **Authors** file.

## Transaction Paging

If your application uses a one-to-many relationship such as Authors and their Books, you may be storing more 'many' records than can appear on a form at one time. There are two keyboard shortcuts for displaying the next or previous 'page' of transaction-type records in this situation:

> CTRL+DOWN   reads and displays the next page of records.
> CTRL+UP   reads and displays the previous page of records.

**Note**       When a Superbase insertion point is active (when it is shown on screen) some of the keyboard equivalents have an editing function, and do not operate as record selection keys.

**Current Record**
After some actions the screen does not show any record data. Click this button to redisplay the current record or to restart Table View with the current record at the top. With a form, the up and down parts of the button select the previous and next pages respectively if the form has more than one page. There are two keyboard equivalents for this button: the UP and DOWN keys.

**Next Record**
Allows you to view the next record or set of related records on a form. The order in which the records appear is decided by the current index, which is shown by the message on the title bar of the main display window. In Table View the data for next record appears

underneath the previous record. In Page View it replaces the previous record. The keyboard equivalent for this button is the RIGHT key.

### Previous Record

This gets the previous record or set of related records on a form, relative to the current record. The keyboard equivalent is the LEFT key.
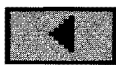
### Fast Forward

Instead of moving forwards through the data one record at a time by pressing the Next button, it is quicker to use the Fast Forward button. In Table View, Superbase gets the next records in the file, displaying them row by row until the screen is full, when it selects the Pause button. The keyboard equivalent is SHIFT+DOWN .

### Pause

Pause is useful during a fast forward or a rewind when you want to stop to read the screen while searching for a record. This button may also be activated by pressing SPACEBAR .
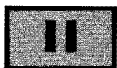
### Rewind

This does the opposite of the Fast Forward button. You will see the data appear from the top of the screen, in reverse index order. If you do this immediately after Fast Forward there will be an apparent delay while Superbase displays records over the text of the existing display. For the keyboard equivalent, press SHIFT+UP .

### First

Pressing this button automatically displays the first record or set of related records on a form, according to current index order. The keyboard equivalent is SHIFT+LEFT .

### Last

Pressing this button automatically displays the last record or set of related records on a form, according to current index order. The keyboard equivalent is SHIFT+RIGHT .
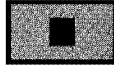
**Stop**

If you want to stop the Fast Forward or Rewind display, click Stop or press the key combination CTRL+C . You must also click Stop if you want to interrupt Pause when it's selected. To stop a process, use CTRL+C .

**Note**  You may turn off the browsing controls display from the System Options requester, invoked from the Set menu. All browsing, key lookup and filter controls are available from the keyboard.

# Key Lookup

**Key lookup button**

Use the Key Lookup button for the retrieval of individual records. You enter a value for the field on which the file is currently indexed, and Superbase uses the index to find and display the record. You can only use one index field at a time – the Filter button is used for multiple value searching. With a form, a key lookup retrieves the record or set of related records using a value in the current index of the current file, as indicated in the window title bar. To select Key Lookup from the keyboard use the '?' key.

When you click the Key Lookup button you'll see a requester asking for the entry of a key value. The maximum length of a key value entry is 100 characters.
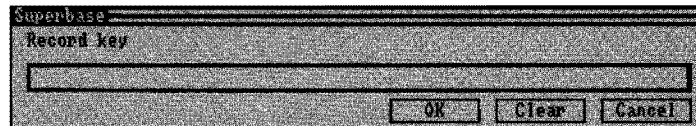
## Retrieving a Record by Key

This procedure is best illustrated by an example. Suppose you wished to look up the record for Ms Julie Kasper in your **Clients** file using her reference number. First you would select the **Customer_ref** index by pressing the '-' key. Then you would click the Key Lookup button, type in the key – KASP00017 – and press ENTER .

When you press ENTER or click OK, Superbase looks up the first record for which the **Customer_ref** field has the entered value. If there were more than one with this value, you would have to browse through the file with the Next Record or Fast Forward button to find the exact record you wanted.

You are not restricted to one index. Superbase allows you to have up to 999 indexes for each file, all automatically updated when you add or modify records.

## Entering Partial Values

You do not need to type the full index key in every case. For example, you could retrieve the record for Herr Huttermann by entering just the three characters 'Hut'. All you need do is enter enough characters to identify the record you want.

When the key you enter does not exactly match any record in the file, Superbase informs you that it selected the next record in index sequence.

# Filter
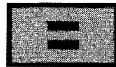
**Filter button**

 This is a very powerful feature. You may have thousands of records in a database file, yet you may often want to review them (or print them out) on a selective basis. The Filter allows you to enter a set of values, as simple or as comprehensive as you like, to restrict the display to only the records that match those values. On a form, each file may have its own filter.

Pressing the '=' key has the same effect as clicking the button.

## The Filter Requester

Move the pointer and click the Filter button on the browsing controls.
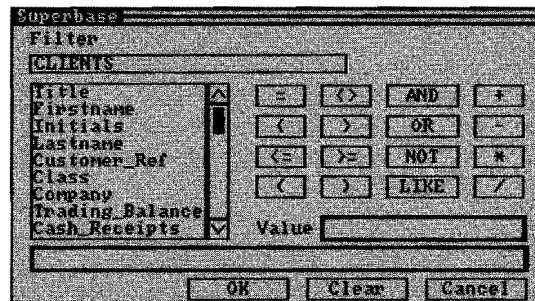
The requester has three main parts: the Fields box, the operators and Value box, and the Text box. A Filter is created by a combination of clicking on field names and operators, and typing in values. These actions produce the Filter command line in the Text box. Let's look at each component of the Filter requester in turn.

## Fields Box

This appears on the left-hand side of the requester. You can scroll the list of field names up and down in the usual way, using the scroll bar and arrows at the right of the Fields box. You specify that a field is to be used in the Filter by clicking on its name in this panel.

The list of field names is complete. It is not affected by whether you have a selected fields list, so the fields that specify the Filter need not be the same as the ones that are shown in the Work Area.

## Operators

These appear to the right of the Fields box, and can be divided into Relational operators, Logical operators and Mathematical operators. We'll see how these are used below.

## Value Box

You use this box for typing in values such as names, numbers, and dates, which will form part of the Filter. When you are matching numeric data that has to be held in a text field because it may include non-numeric characters (such as telephone numbers), you must put the value in quotes.

Remember to press ENTER or click the Value button after typing in the Value box. You must do this to ensure that the contents are copied to the Text box.

## Text box

This is the long box that extends across the whole requester. When you make selections with the mouse or type values into the Value box, Superbase copies them into the Text box, building up the Filter command line. In this way you get immediate feedback about the results of your selections. The maximum length of the Filter command line is much greater than the apparent length of the box – up to 255 characters are allowed. You can type directly into the Text box if you prefer.

## Using the Operators

The operators are used to relate field names to other fields or values. For example, if we wanted to set up a Filter to show only the addresses in West Germany, we would need the expression: **country** LIKE "West Germany".

Here, **country** is a field name obtained by clicking in the Fields box; LIKE is selected by clicking on the LIKE operator button; and 'West Germany' is typed into the Value box (note that it is not necessary to enter quotation marks if you use the Value box).

## Text String and Pattern Matching: LIKE

When you're searching for names or text strings, you should avoid using the equals sign operator, and use LIKE instead. This is because only LIKE accepts matches on either upper or lower case characters; it's said to be 'case insensitive.' The equals sign operator, on the other hand, requires exact matches for what you type in: it is 'case sensitive.'

LIKE also permits searches for text strings using pattern matching characters. These allow you to define text strings with fixed patterns of characters whose content may vary, or text strings of indefinite length as well as content. In addition, you can also use square brackets to define specific ranges of characters to be accepted by the Filter.

## LIKE Syntax

| | |
|---|---|
| LIKE "*" | Searches for empty fields with '*' on its own |
| LIKE "?*" | Searches for fields that are not empty |
| LIKE "text" | Field with content 'text' |

The asterisk represents a text string of any length:

| | |
|---|---|
| LIKE "text*" | 'Text' string at start of field |
| LIKE "*text" | 'Text' string at end of field |
| LIKE "*text*" | 'Text' string anywhere in field |

The question mark represents a single character:

| | |
|---|---|
| LIKE "???" | Fixed length, any content |
| LIKE "?text?" | Fixed length, fixed content |
| LIKE "?text*" | Fixed leading length/content, any length |

A range inside square brackets may be used in place of a question mark character:

| | |
|---|---|
| LIKE "[a-m]" | Single character range 'a' through 'm' |
| LIKE "[adg-j]*" | Character 'a', 'd', or 'g' through 'j' |

The caret or up arrow at the start of a bracketed sequence signifies exclusion:

| | |
|---|---|
| LIKE "[^n-w]" | Excludes characters 'n' through 'w' |

You can combine different elements of the pattern matching syntax:

LIKE "a??[c-h]?x*"

This matches 'a' in the first position, any character in the second and third positions, the range from 'c' to 'h' in the fourth position, any character in the fifth position, 'x' in the sixth position, and any number of characters following the 'x'. LIKE is the operator you should use when searching for text strings in fields.

LIKE "|text"      Soundex searching

Soundex searching finds text strings that 'sound like' the given string. Pattern matching characters may not be used.

## LIKE Examples

Each of the examples uses the demonstration **Clients** file.

**Example 1**

Suppose you want to find the records for the clients whose firstname is Robert. You would have to enter the following in the filter command line box:

Firstname LIKE "robert"

Notice that we don't trouble to enter an upper case 'R', as the LIKE operator is case insensitive. You can try this out now if you wish. Once you have entered this filter command line, click OK. Superbase will then display all the records where the **Firstname** field is LIKE 'Robert'.

**Example 2**

Supposing that you don't know the full first name you're looking for, but you know that it starts with an 's'. Again you can use the LIKE operator to help you. You would need to create the following filter command line:

Firstname LIKE "s*"

The '*' is important, because you are telling Superbase to find any record that starts with an 's' and can be followed by any number of characters.

**Example 3**

What if you don't know which letter the first name starts with, but you know that it contains an 'r' somewhere in the spelling? To find the record this time you would have to use the following filter:

Firstname LIKE "*r*"

**Example 4**

Although this is a slight digression, it's useful to be able to NEGATE a LIKE expression. If you want to get all those first names which don't contain an 'r', you can change the previous filter command line to read:

NOT (Firstname LIKE "*r*")

**Example 5**

Taking our example a step further, supposing you want those records where the fourth character of **Firstname** is an 'r'. In order to do this you would use the following filter command line:

Firstname LIKE "???r*"

**Example 6**

Suppose you don't know any of the letters in **Firstname**, but you know it contains five characters. You could find the record with the following filter command line:

Firstname LIKE "?????"

If you want to find those records where **Firstname** is at least five characters long, you can type an '*' after the five question marks.

**Example 7**

LIKE also allows you to search for ranges of characters. Suppose you want to list just those records where **Firstname** begins with 'a', 'b', or any letter between 'j' and 'm'. The filter command line should read:

Firstname LIKE "[abj-m]*"

Notice that whatever is inside the square brackets represents a single character. If you want to specify ranges for more than one character in a string, you must use a set of square brackets for each one.

## The Relational Operators

| | |
|---|---|
| = | This specifies that the field is to be EXACTLY EQUAL TO the following value or field: Balance = 2000 |
| <> | The field is to be NOT EQUAL TO the following value or field: **country** <> "France" |
| > | The field is to be GREATER THAN the following value or field: Balance > 1000 |
| < | The field is to be LESS THAN the following value or field: Balance < 1000 |
| >= | The field is to be GREATER THAN OR EQUAL TO the following value or field: Balance >= 1000 |
| <= | The field is to be LESS THAN OR EQUAL TO the following value or field: Balance <= 1000 |

## The Logical Operators AND, OR and NOT

These provide links between different expressions like the ones given above. You use them to specify alternatives and ranges for the Filter, either for a single field or for a number of interrelated fields:

**AND**      Both of the conditions linked by AND must be satisfied for the record to be accepted by the Filter: **country** LIKE "West Germany" AND Balance >= 5000

**OR**      Either of the conditions linked by OR is sufficient to make the record acceptable to the Filter: **country** LIKE "West Germany" OR **city** LIKE "Geneva"

**NOT**      NOT is slightly different. It negates the effect of an expression: NOT (**country** LIKE "West Germany" AND Balance > 5000); this will find all the records other than the ones where the country is West Germany and the balance is greater than 5000. Note the use made of parentheses with NOT.

## Parentheses

The '(' and ')' operators are used in the normal way, to give priority to one part of the expression over another.

(      The following expression has a higher priority for evaluation than the preceding expression.

)      Required to end a higher priority expression.

The numbers of opening and closing parentheses must be equal.

## The Mathematical Operators

+      This is used to express addition:
Balance > Amount + 1000

−      The minus sign expresses subtraction:
Balance > Amount - 1000

*      The asterisk is used for multiplication:
Balance > Amount * 1.15

/      The slash represents division:
Balance < Limit / 10

## Typing in the Text Box

The great power and flexibility of the Filter system is only achieved by giving the user control of every aspect of the operation. Because you can type into the Text box you can include all kinds of expressions which are not available from the buttons on the requester itself, and all kinds of text and numeric values.

The price you pay for this flexibility is an increased risk of making mistakes. This is not too serious, as you cannot harm your data by using a Filter for browsing (although you can delete records using the Process Remove filter).

However, you may see a Superbase error message concerning an error in the Filter command line in two possible situations:

- When you click OK after creating or editing a Filter command line.

- When you use a Filter after changing the definition of a field with Project Modify File.

If you see an error message you must either change the Filter command line or change the file definition. Here's a summary of the editing facilities available when typing inside the Text box:

## Filter Command Line Editing

You can remove everything from the Main Box by clicking on the Clear button or, if you've already clicked in the box, by pressing AMIGA+X .

- The DEL key removes the character to the right of the cursor. Holding the key down repeats its action.

- The Backspace key removes the character to the left of the cursor. Holding the key down repeats its action.

- The keys SHIFT_LEFT and SHIFT+RIGHT   move the cursor to the beginning and end of the filter command line respectively.

- Provided the cursor is still present, AMIGA+Q   will undo any changes you have made.

## Using a Filter

The Filter command line is made active by clicking on OK. Whenever a Filter is in use, Superbase highlights the Filter button. You can perform all but one of the browsing operations we've discussed, and Superbase will restrict the records displayed to the ones that match the Filter values. For example, if you have specified **country** LIKE "West Germany" and you click the First Record button, Superbase will find the first record in the file, according to the current index, which has the value 'West Germany' in the **country** field.

The Key Lookup button is an exception to the Filter effect. You can still recall individual records by entering an index key value, whether or not they match the Filter.

## Suspending a Filter

There will be times when you want to remove the Filter effect so that you can review all the records in the file again. To do this, just click the highlighted Filter button to turn off the highlight.

## Modifying or Re-activating a Filter

If you've set up a Filter and then suspended it, you can bring it back into operation by clicking once more on the Filter button. Superbase displays the Filter requester, with the Filter command line in the Text box just as you left it.

If you want to, you can add to the Filter command line by clicking on more field names and operators, or edit it after clicking in the Text box, as described above.

Whether you've made any changes or not, just click OK to use the Filter again.

## Optimizing the Filter

By using a Filter, you can search for data in any field whether it is indexed or not. However, if it can make use of an index, Superbase will retrieve records far more quickly.

Superbase uses an index to optimize Filter operations when the following conditions are satisfied:

1.  The index field occurs at the beginning of the Filter command line.

2.  The index for that field is the current index (use the '–' key on the numeric keypad to change the current index).

3.  The first operator to occur in the command line is a relational operator; i.e., it must be one of the following: =, <=, >=, <, >. The Filter operation will not be optimized when LIKE is used as an operator.

## Filters and Forms

Each file on a multi-file form may have its own filter command line set. When you browse through the data, Superbase only retrieves records that match all the filters.

Although you cannot store filters as re-usable disk files in themselves, you may assign a filter to a function key and store the function key set on disk. See Chapter 29 Function Keys for further information.

# 11 WORKING WITH DBASE FILES

The Process Import command allows you to convert dBase files to Superbase format (see Chapter 21 Import). However, you may just want to read or display dBase files without modifying their data. For this kind of application, Superbase can act as a front end to dBase. It enables you to open dBase files without converting them first.

dBase files are treated as Read Only files. You cannot edit a dBase file's data or write to the file. Apart from this, you may use a dBase file in any of the following Superbase operations:

- Printing. You can show dBase data on a Superbase form, created with the Superbase Form Designer graphic design system, and print high quality graphics output.

- Browsing. Select records using the browsing controls with or without a filter.

- Key lookup. Retrieve a dBase record using an index key or by record number.

- Mailmerge. You can extract data from a dBase file and merge it with a Superbase Text Editor document.

- Query. dBase files may be used in all standard Query operations.

- Export. You may export a dBase file using any of the Export file type options, using a subset of fields and applying a search filter if you wish.

## Opening a dBase File

You open a dBase file in a similar way to opening a Superbase file:

1. Select Open and dBase File from the Project menu. Superbase uses '*.DBF' as the pattern matching string and displays a list of the dBase files in the current directory.

2. Click a file name and, when it appears in the Text box, click OK (or just double-click the file name).

The dBase file will now be read into memory and displayed on screen in the same format as a Superbase file. If Superbase finds a field name which is used as a reserved word (see Appendix C for a list of Superbase reserved words), it adds an underscore character to the name.

## Using an Index with dBase Files

Initially, Superbase displays dBase records according to the order in which they are stored on disk, and it does not use an index. If you select the Key Lookup function, it asks you for a record number rather than an index key.

To open a dBase index:

1. Choose Open and Index from the Project menu. Superbase displays a list of NDX files.

2. Select an index in the normal way – by double-clicking the index file name, or clicking the name once and then clicking OK.

3. Repeat steps 1 and 2 for any other dBase indexes you may wish to activate.

If the dBase index applies to a single field, Superbase treats it in the same way as its own index files. The Key Lookup button and other browsing controls now follow the index sequence; where possible, filters and queries are optimized on the current index.

If the index is a composite index which applies to more than one field, Superbase displays a confirmation message requester. This shows the fields which the index applies to and warns that 'This composite index may not be valid.' Click OK to proceed. You will be able to use the composite index for browsing operations, but it will not be used to optimize a filter or query.

**Note**      If you have opened a number of dBase indexes, you can use the '–' key on the numeric keypad, or the Open Index command on the Project menu, to change the current index. Pressing the '–' key selects each of the indexes in turn. It also offers a 'no index' option, allowing the user to browse through a file using the sequence in which records are stored on disk.

## Saving a dBase File Definition

When you select the Save dBase File option on the Project menu, Superbase saves its own internal representation of the current dBase file definition as a DBD file. The DBD file stores the details of the file structure and any indexes which are associated with the dBase file, just like a Superbase SBD file.

It also stores the locations of the fields in Page View, so you have full control over the screen design.

The next time you open the file, Superbase will automatically make use of its DBD file and its indexes will be immediately available.

## Modifying Field Display Formats

You may use the File Modify command to change certain display formats. Superbase offers a wide range of numeric and date display formats, and you can apply these to the fields in a dBase file without affecting the dBase data in any way.

## Creating Virtual Fields for dBase Files

Although you may not write to dBase files from within Superbase, you may add Virtual fields to the file definition (DBD extension) with the Project Modify File command. Virtual field data only exists in memory, not on disk.

This feature allows you to perform 'what if' operations on dBase data, by specifying a Calculation formula for the Virtual field that uses data already in the file. The Virtual field may then be included in list and query output.

dBase file definitions may also be used to apply Superbase numeric and date formats not available in dBase itself to data in the dBase file.

## LOOKUP with dBase Files

Superbase cannot automatically open the file definition for a dBase file (DBD extension) from a normal file definition (SBD extension) that contains a LOOKUP to the dBase file. You must open the dBase file separately if you plan to use it as a lookup file.

## dBase Files in Forms

You may use dBase files on forms created with the Form Designer, provided you create a DBD file first, as outlined above. dBase files accessed from a Superbase file remain Read Only.

# 12 SELECTING VIEWS AND FIELDS

Superbase gives you precise control over the way record data is presented on screen. You may choose to use a form to display data. The Form Designer allows you to design as many different screen forms for use with a single file as you wish, selecting just the fields that are relevant for each part of your application. Forms may display fields from multiple files, and many applications demand the comprehensive facilities for relational browsing and data entry that the Form Designer provides.

However, if you wish to build your application around single files, you may prefer to use Superbase's default views, which provide excellent presentation facilities at a simpler level.

With the Field Selection command, you can specify which fields are to be displayed. In addition, there are different ways of displaying records: Table View, Page View, and Record View.

## Selecting Fields

As Superbase imposes no limit on the number of fields per record, you need to have a way of restricting the fields that actually appear on the Work Area. You can do this with the Field Selection command on the Set menu.

This is one of the most useful commands in the whole Superbase system. You use it to select a group of fields from the list of fields in the File Definition. These fields then become the default fields for many Superbase functions, including the Work Area. Thus if you have a file with many hundreds of fields per record, you can choose, if you wish, just three or four to work with.

From the Set menu, select the Field Selection command. Superbase displays this requester:
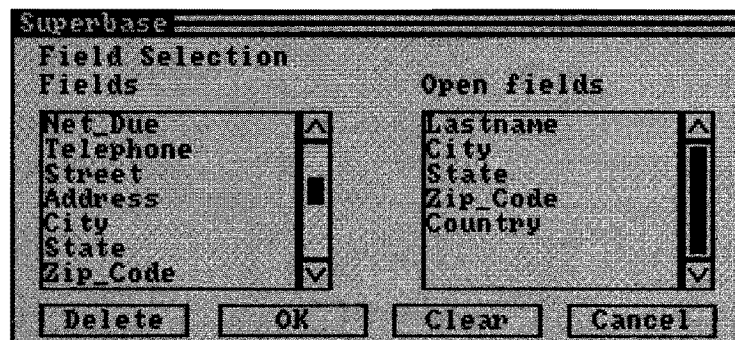


The basic procedure for selecting fields is very simple:

1.  Click a field name in the left-hand box. Superbase copies it into the
    right-hand box. If you make a mistake, Delete removes the current field
    from the right-hand box; Clear removes the whole list and restores all the
    fields in the file for viewing.

2.  Repeat until the selection of fields you want is in the right-hand box.

3.  Click OK.

Here are some tips to keep in mind when using this function:

*   You can select fields in any order.

*   You can repeat fields.

*   You can remove a field from the selected fields list: click its name, then
    click the Delete button.

*   Repeated clicking on Delete removes field names from the bottom of the
    list.

*   To move a field to a new position in the list: click its name in the
    right-hand panel; click Delete; click the field before the desired new
    position in the right-hand panel; click the field name in the left-hand
    panel.

*   To remove the selected fields list from the right-hand panel, click Clear.

When you have selected a restricted set of fields with this command,
Superbase places a checkmark against the command name. If you want to
suspend the selected fields list so you can view the full set of fields, select the
command again.

## The Selected Fields List

The set of fields that are selected with this command are referred to as the
selected fields list. Superbase uses this list in most of the menu commands:

*   The Work Area, Record, Page and Table View.

*   Edit Current, Record New, and Record Duplicate.

*   Process Query, Process Import, Process Export, Process Print, and
    Process Labels.

## Storing the Selected Fields List

The selected fields list for the current file can be saved with File Save
command. It will then become the default fields list when the file is next
loaded. To activate a list after it has been loaded, choose the Field Selection
command and click OK.

# Record View

The default record format which is used when you first start up Superbase is only one of the three possible ways of looking at your file. It's called Record View. This view shows the records one at a time, restricting the format by keeping the field names on the left and limiting you to one field per line.

The data from the open fields for each record is shown in a block. If Paging (described at the end of this chapter) is off, the record blocks will scroll up or down the screen when you use the browsing controls.

# Table View

Table View is used to display or print records in columns. It displays the field names across the top of the screen with the data appearing in rows and columns, each record occupying one line.

While you are using Table View to browse through your records the field names appear at the top of the screen, with the data for each field below the name.

While Table View is selected, you can move the columns of data right or left by clicking and holding down the mouse button in the column you want to move, then dragging the column to its new position. There are some rules governing what you can and cannot do:

- You cannot move the left-hand column.
- All columns to the right of the selected column move with it.
- After moving a column to the right, you may have to redisplay the data if a column has been truncated.
- The current settings remain even if you select a different set of fields for the file.
- If you have dragged columns to compress the display, you can restore the true length of a column by double-clicking in it.

# Page View

Page View allows you to design your own screens for entering and displaying record data.

The first time you select Page View for a new file, the field names are displayed down the left hand side of the screen, with the data for the current record, if there is one, to the right, one field per line.

You can rearrange the screen. You do this by clicking on a field name, and, while keeping the button depressed, dragging the field to its new position on the screen.

There are a number of rules governing the way fields are displayed in Page View:

- The worksheet on which the fields can be positioned is 273 columns wide.
- A maximum 256 lines are available. The vertical scroll bar reflects the lowest field position.
- Fields which are not selected are not shown.
- Paging is automatically switched on. This allows you to scroll the window in both axes.
- You cannot overlap field names or data areas.
- Only the open fields are shown, so close the selected fields list if you want to see all the fields in Page View.
- You must save the File Definition with the File Save command in order to store the current Page View.

If you're in Table View and you select Edit Current, Record New, or Record Duplicate, Superbase automatically switches Page View on.

### Scrolling the Screen

While you are viewing your records in Table View you will notice that the field names do not all fit onto the screen. The fields to the right can be viewed using the arrow at the right of the bottom scroll bar. Move the pointer onto this arrow, and click it once. The display shifts five columns to the left. Click once on the left-hand arrow to shift right and restore the display. If you want to shift the display by larger areas, drag the bottom scroll bar with the mouse, or click in the scroll box.

# Switching Between Views and Form Display

The Set Form command is ghosted unless a form is open. When you open a form, a checkmark is placed against Set Form, and removed from the view that was active before the form was opened.

However, the default views are still available when a form is open. If you select one of the view commands, the form display is replaced by a single file display of the current file.

You can return to the form display, provided you have not used the Project Close Form command in the meantime, by selecting the Set Form command.

# Show Field Names

With the Show Field Names command on the Set menu, you can specify whether field names are displayed or removed from the screen. It applies to the way records are displayed in all three view modes. By default, Superbase shows the field names and places a checkmark against the menu command. You toggle the command on and off by selecting it from the menu in the normal way.

The Show Field Names command also applies to printed output. If you output records to the printer using the Project Print command, the field names will not be printed when Show Field Names has been turned off.

# Paging

While you are viewing the records within a file you may only want to see one 'page' at a time. The effects vary depending which of the three Views described above is switched on. You toggle Paging on and off by selecting it from the menu in the normal way.

## Paging in Table View

When Paging is on, Superbase displays a screen of record data, one record per line, until the screen is full. It then selects the Pause button and waits. Click Pause to see the next screen of data, or another record selection button to perform another action.

When Paging is off, Superbase continues until it reaches the end of the file or you interrupt it by clicking Pause or Stop.

## Paging in Record View

When Paging is on, Superbase displays one record at a time on a screen by itself. When paging is off, Superbase displays the records one after the other on the same screen. Fast Forward or Rewind scrolls through the records until Superbase reaches the end of the file or you interrupt with Pause or Stop. (If a record is longer than the screen then on single record selections such as Next Record Superbase scrolls to the last field and stops.)

Paging in Page View is always switched on.

The Paging function does not affect the printer.

# 13 PRINTING DATA AND FORMS

The Print command in the Project menu allows you to select what to print, and, as examples, the options to print the current record and the current file are described in this chapter. Both options may be used with or without forms. See Chapter 14 Defining and Using Queries for information on printing queries. See Chapter 31 Customizing Your System for information on printing the Status screens.

The appearance of any printed output, however, depends upon the features available in your printer and Superbase's ability to utilize those features.

You are supplied with several printer control files (such as S:SB_EPSON.INI, S:SB_LASERJET_III.INI and so on) which contain the details Superbase needs to operate various printers, and before you print from Superbase for the first time you should run the Printer Setup command in the Set menu to specify to Superbase the type of printer you intend to use.

Most printers, in addition to their normal typeface, can print in underlined, italic or bold typefaces. Some printers have various fonts available, such as Courier, Times Roman, Univers and so on, and possibly several sizes of each font. The Printer Setup command allows you select the font and size from all those available on your printer, and you can specify underlined, italic or bold typefaces where required in the output from queries and reports.

## Page Setup and Font Selection

If you are using a form, the page dimensions and font selection are taken from the form itself. However, if you intend to print single records or lists you should use the Printer Setup command on the Set menu to define the font, the font size and the page margins for printed output. Selecting Printer Setup from the Set menu first displays a Change Printer requester like this:
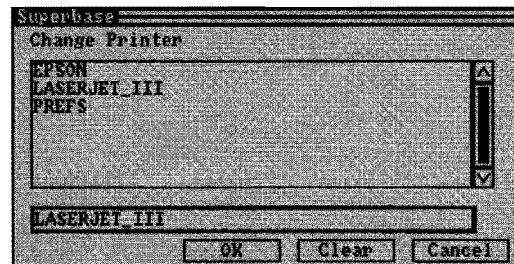


A list is displayed of all your printer control files. You may need to scroll the list to see all of the names. If any printer control file is currently selected, its name appears in the one-line box below the main box.

What you have to do is to ensure that the printer control file selected matches the printer that you intend to use, or corresponds to a printer with which your printer is exactly compatible. If you need to set for the first time or to change the name of the selected printer, point to the name you want in the list and click the mouse button. When you are sure the correct name is selected (displayed in the one-line box), click on OK.

If no displayed name quite matches your printer, see Creating a New Printer File, later in this chapter.

One additional filename appears in the list of printer control files – it is PREFS. This refers to the printer specified in Workbench Preferences. You may select the PREFS printer if you wish, but in the Page Setup requester (described below) you will be able to alter only the Initialize field.

When you have selected your printer, the Page Setup requester is displayed, looking similar to this:
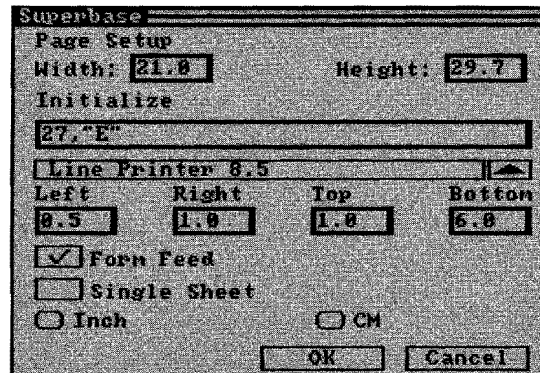


1.  Examine, and change if necessary, the values set for the width and height of the paper that your printer uses. These are the overall dimensions of a sheet of paper, not the area within which you want to print. You can specify the size in either inches or centimeters, depending on which of the buttons near the bottom of the requester marked 'Inch' and 'CM' is selected.

2.  Specify the left, right, top and bottom margins, again using either inches or centimeters as specified by the 'Inch' and 'CM' buttons.

3.  If your printer will start a new page after being sent a 'form feed' character (ASCII 12), then click the Form Feed button so that it is checked. If your printer does not correctly handle a form feed character, leave this button unchecked and Superbase will send as many blank lines as necessary at the bottom of each page to ensure printing starts at the top of the next page when required (although this method is not quite as reliable as sending a form feed).

4. Click the Single Sheet button so that it is checked if you need Superbase to pause at the end of each page of printed output (so that, for example, you can load special stationery one sheet at a time).

5. If the printer has a variety of fonts and/or sizes available, the currently selected font and size are shown in the one-line box that has an upward pointing arrow at its right. Click on the arrow repeatedly to cycle through all the available fonts and sizes, and stop when the combination you want is displayed.

6. The remaining one-line box is the 'Initialize' box. This contains the codes that Superbase must send to the printer whenever it needs to be sure that the printer is in a known state (at the top of a new page with underlining, italic and bold all switched off, for example). You should not need to change these values unless you are creating a new printer control file.

**Note** Although Superbase does not support proportional fonts on screen (except on forms), you may select a proportional font for printing using the Printer Setup command. If you do use a proportional font you should avoid instructing Superbase to position printed output according to 'column' number.

All the Superbase Print commands – in the Project menu – adopt the settings that are specified with Printer Setup. Whenever you modify the printer settings, Superbase saves the name of the selected printer control file in the S:SUPERBASE.INI file. So if you intend to use the same format for all your printed output, you only need to select the Printer Setup command once. If you wish to define a different format for each type of print operation – for example: printing single or multiple records, printing in different view modes, or printing status information – you should use Printer Setup before selecting one of the print commands.

# Creating or Modifying Printer Control Files

In the majority of cases one of the supplied printer control files will enable Superbase to exploit the features of your printer correctly. If, however, your printer is not supported by one of the standard control files you should be able to create a suitable printer control file of your own.

All Superbase printer control files have names of the form:

S:SB_printername.INI

where printername is, for example, EPSON or LASERJET_III . You should not change any of the supplied printer control files, in case you later need them in their original form, but you should use one of the existing files as a model because you will probably need to make only minor changes.

To create a new printer control file, use a text editor. Superbase's text editor is one that is suitable for this purpose and its use is described in Chapter 23 of this manual. Run the text editor, read into memory the existing printer control file that is nearest to the one you need, make the necessary changes, and save the results in a new file with a name of the form given above.

As an example, the following listing shows typical contents of a S:SB_LASERJET_III.INI   printer control file:

```
Fontname=Line Printer 8.5
Width=21.0
Length=29.7
Initialize=27,"E"
Left=0.5
Right=1.0
Top=1.0
Bottom=6.0
Ffeed=1
Ssheet=0
INCM=0

Font=Courier 12 pt
Plain=27,"(8U",27,"(s0p10.00h12.00v0s0b3T"
BF ON=27,"(s3B"
BF OFF=27,"(s0B"
UL ON=27,"&d0D"
UL OFF=27,"&d@"
IT ON=27,"(s1S"
IT OFF=27,"(s0S"
CPI=10
LPI=6
SETCPI=
SETLPI=27,"&l6D"

Font=Line Printer 8.5
Plain=27,"(8U",27,"(s0p16.67h8.5v0s0b0T"
BF ON=27,"(s3B"
BF OFF=27,"(s0B"
UL ON=27,"&d0D"
UL OFF=27,"&d@"
IT ON=27,"(s1S"
IT OFF=27,"(s0S"
CPI=16
LPI=8
SETCPI=
SETLPI=27,"&l8D"
```

Most lines start with a keyword, then have an equals sign and finally a series of one or more values, separated by commas. The empty lines that separate the groups are ignored by Superbase.

The first group of lines specify values that apply whenever this printer control file is selected. They show which font is currently selected on this printer, and the current values of all the items that you can set with the Page Setup requester (described above).

Each subsequent group of lines refers to a particular font. Only one of these groups of lines will apply at any one time – the group having the font name currently specified in the Fontname line at the beginning of the file.

The first line of each group of lines after the first identifies the font name, and is followed by lines which specify the codes that Superbase must send to the printer in order to switch certain features on or off. The keywords used and their meanings are as follows:

| Keyword | Meaning |
| --- | --- |
| Plain | Boldface, underline and italic off |
| BF ON | Boldface on |
| BF OFF | Boldface off |
| UL ON | Underline on |
| UL OFF | Underline off |
| IT ON | Italic on |
| IT OFF | Italic off |
| CPI | Pitch (characters per inch) |
| LPI | Line spacing (lines per inch) |
| SETCPI | Reset CPI value |
| SETLPI | Reset LPI value |

Each line but the first, in the second and later groups, has a series of values after the equals sign. This series of values represents a series of bytes to be sent to the printer, and bytes may be specified in any one of three ways: individual decimal ASCII values, individual hexadecimal ASCII values, or a text string. For example the string 'Xyz3' may be specified as any one of:

```
"Xyz3"
88,121,122,x33
x58,x79,"z",51
```

To find the series of bytes needed to perform each of the above actions on your printer, refer to your printer's technical reference manual. Each series generally begins with an ESCAPE character (ASCII 27), and these series of bytes are often referred to as 'escape sequences'.

# Printing Forms

Forms designed with the Form Designer may be printed from within the database. You can achieve a high degree of page fidelity to the screen, but the differences between fonts on different printers and between screen and printer fonts can lead to some discrepancies between what you see and what is printed.

You print forms with the same commands used when printing without forms: Print Record and Print File, both in the Project menu. They are described below.

## Form Fonts

If you are designing a form that will not normally be printed, such as a specialized data entry screen, you should use screen fonts for all text.

If you intend your form to be printed, and you know that the same printer will always be used, it is best to design it with fonts offered by the driver for that printer. The Form Designer and Superbase will agree about the nearest font to use for screen display.

However, if your form may be printed on different printers, for example if you are designing a form for use by several individuals in your organization, you should try to use fonts that are offered by most printers, such as Roman, Script or Modern. Even so, minor differences in fonts designed by different printer manufacturers may cause small problems when printing text.

# Printing Views

Before printing records with the Print command (on the Project menu), you should select the view mode you require using the Set menu. Superbase prints records in the format of the current view mode:

- In Table View, the field names appear as column headings, separated from the first line of data by a blank line, on each page. If the field data and names extend beyond the right margin, they are wrapped onto the next line. Superbase then inserts a blank line before printing the next record.

- In Record View and Page View, records are not split over two pages; if there are not enough lines at the bottom of a page to print a record, Superbase skips to the top of the next page. Record data that extends beyond the right margin is truncated.

- In Record View, Superbase prints a blank line between each record. In Page View, it follows the blank line with a line of dashes.

### Show or Hide Field Names

If field names are displayed on screen, they will also be output to the printer. Before printing records, you can use the Show Field Names command on the Set menu to specify whether the field names are to be printed or not.

- In Table View, the field names appear as column headings, separated from the first line of data by a blank line, on each page. You may need to drag columns to the right on screen to set a width sufficient to avoid truncation when printing in some proportional fonts.

- In Record and Page View, the field names, which are shown on screen in reverse video, are printed normally.

# Single Records and Forms

The Print Record command on the Project menu prints the current record or form. When you want to print a single form, follow this procedure:

1. Open the form with Open and Form on the Project menu.

   ❑ The form opens any files it needs.

2. Select the data you want to print.

3. Select Print and Record on the Project menu. The Print Options requester appears:
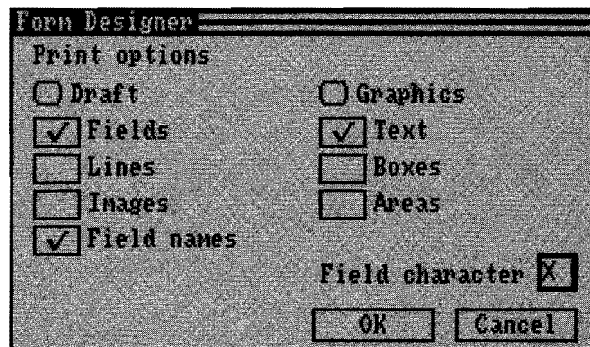


Superbase only invokes the Print Options requester when you have a form open. It allows you to choose whether to print the current page or the whole document if your form has more than one page.

You may also choose which classes of graphic objects are to be printed. Areas, boxes, lines, text, and images on forms will all be printed unless you turn off the relevant check box.

Make the selections you want and click OK.

You should obtain a reproduction of the screen form on the printer.

- Remember that form objects designated as non-printing – such as operator prompts – will not be printed.

If you are printing without a form open, simply select the record to be printed and then choose the Project Print Record command. The command makes a copy of the record as it appears on screen in the current Page or Record View. As well as the contents of the record, it also prints field names if they are shown on screen.

Unless you are printing a form, Project Print Record uses the printer and page settings that are specified with the Set Printer Setup command.

# Printing the Current File

The Print File command on the Project menu provides a quick and easy way of obtaining a printout from the current file. This command should also be used when you want to print a series of forms.

If you have a form open, Print File outputs data using the form. If you are not using a form, Print File outputs the currently selected fields list in the format of the current view mode, using a standard Filter. If field names are shown on screen, these are printed as well as the field data. Records are printed in the order of the current index. The Print File command displays a Print Order requester which allows you to to change the current index, and to select either ascending or descending order.

You may also make selections from the Print Options requester (see above).

**Note**       Make sure that your printer is connected and is on-line before you commence any printing.

## Using Project Print File

1.  Select Print and File from the Project Menu.

2.  Superbase displays the Print Filter.
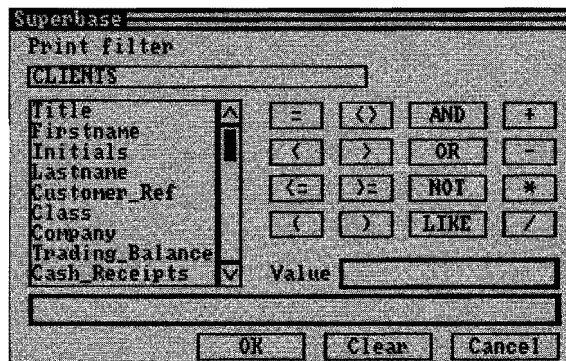
❑   The requester is the same as the normal Filter requester. See
     Chapter 10 Retrieving Data Interactively for a detailed explanation
     of how filters work.

❑   The filter applies to the current file only, even when a multi-file form
     is in use. Filters defined for files other than the current file are used.

3.   Build up your Filter command line and click OK.

4.   Superbase displays the Print Order requester. By default, records are
     printed in the order of the current index. If you wish to change the index,
     you can select another index from the List box.

5.   Click in the Descending check box, if you wish the records to be printed in
     descending order. Then click OK.

6.   Select Single Page or Whole Document from the Print Options requester,
     and turn off any class of graphic object you do not want to print. Click OK
     to print the form.

Unless you are printing a form, the Process Print command uses the printer
and page settings specified with the Set Printer Setup command.
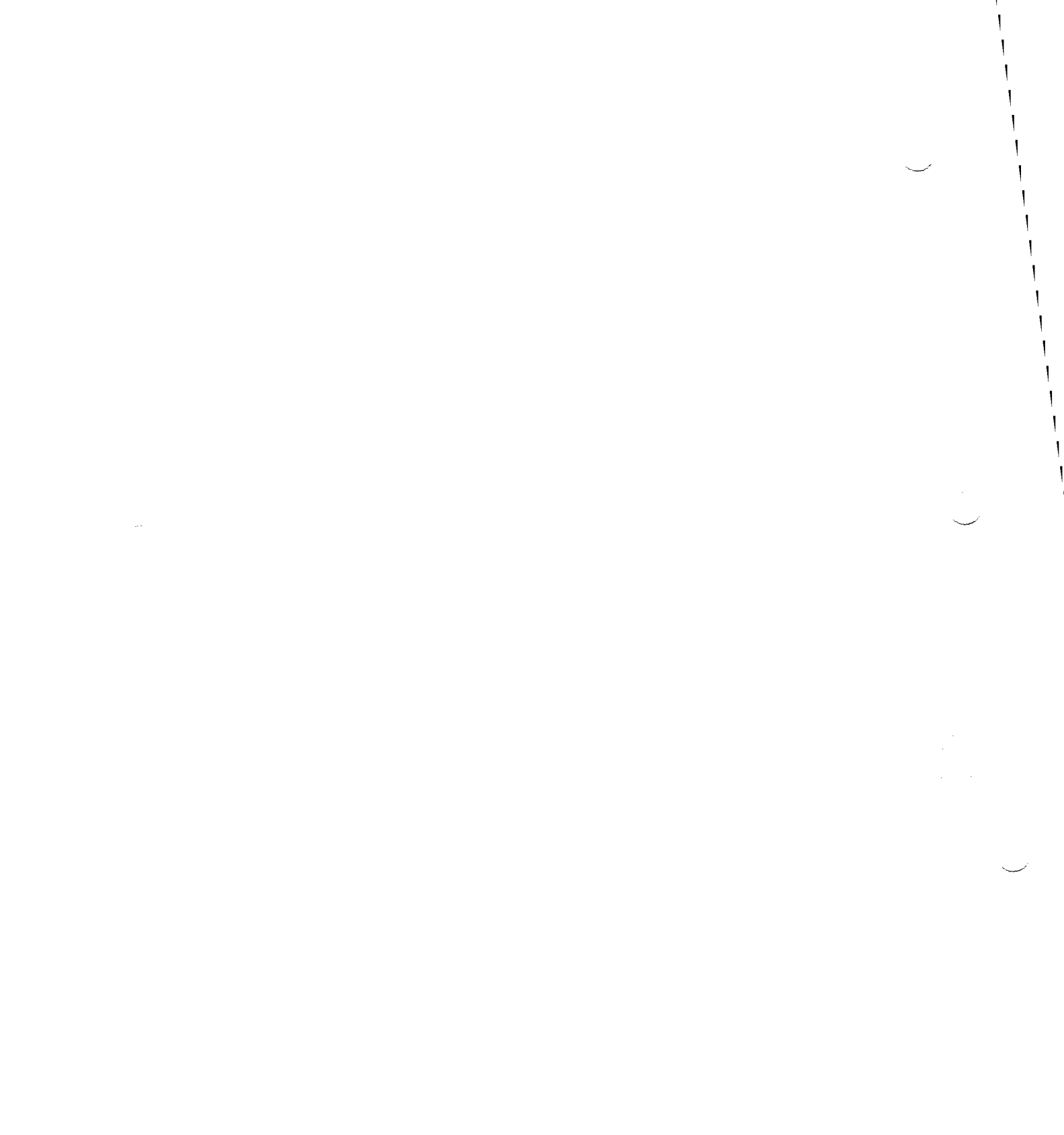
# 14 DEFINING AND USING QUERIES

The Query command is a powerful and versatile tool for selecting data. In this chapter we explain how to define single file queries, focusing on basic concepts and operations. See also Chapter 15 Reporting and Chapter 17 Multi-file Applications.

Superbase's DML offers a more powerful query facility in the form of a multi-line SELECT statement. This allows you to construct queries with more output fields, search conditions, and sorting levels than are possible in the Query Definition requester. The SELECT statement may also be combined with a set of Reporting commands to produce fully formatted and analyzed reports. The Form Designer's Report Generator creates report programs that make use of both SELECT and the Reporting commands.

For many users, though, the Process Query menu command provides all the functionality they require. If you use this command to build up a library of re-usable queries on disk, you may still run them later from within a larger DML application.

## Query Applications

Query is the most sophisticated of Superbase's menu commands. In some ways it is similar to the Filter facility available from the browsing controls and in Project Print. But unlike the Filter, Query allows you to define not just one, but four command lines: Fields, Report, Filter and Order lines. Each line is used to specify a different aspect of the query output. It allows you to produce formatted output with a great degree of control over both the selection and the presentation of record data.

Query is a multi-purpose command which has five main areas of application. It is used for:

1. **Creating complex filters**. In many applications, you may find that you need to retrieve the same set of records over and over again; you may need to check groups of records on a regular basis using the same or similar selection criteria. If you used the browsing controls filter you would have to enter the filter conditions in every session. One of the advantages of using Query is that you can save a query on disk.

2. **Multi-file applications**. By setting up a relational link in the Query Filter, you can select data from several files at the same time. See Chapter 17 Multi-file Applications.

3. **Reporting**. The report features available in Query provide additional information about the results of a search, such as record counts or field totals. See Chapter 15 Reporting.

4. **Sorting**. Query output can be sorted into any order. Indexing already provides one way of sorting a file, but with the Query facility you can sort records in either ascending or descending order. You can also specify several levels of sorting, using a different field for each level.

5. **Output redirection.** The output from a query can be directed to one of four possible destinations: the screen, the printer, a text file on disk, or a new database file. This last output command lets you create a new database file which combines the data from a number of existing files.

## Opening a Query

Superbase queries are re-usable. They are stored on disk as SBQ files, which contain the information you define for the four parts of the query, plus the names of all files involved in the query. Even if the query you want has not been defined and stored, a similar one may exist, and it is probably easier to read that one and change it than to start from scratch.

To open an existing query:

1. Select the Query Open command from the Process menu.

2. Select the required query from the list.

   ☐ If the left-hand panel is empty, there are no queries in the current directory.

3. Click OK.

   ☐ Any file involved in the query that is not already open is opened automatically at this point.

Superbase opens the query and presents you with the Query Definition requester. If you want to run the query immediately, just click OK or press ENTER .

See also Saving a Query at the end of this chapter.

Once you have defined a query, that query definition remains available until you define another one or end the Superbase session. To rerun your last query, or to change it before rerunning it, or to define a new query:

1. Select the Query Edit command from the Process menu.

   ☐ The last-used query definition is displayed in the Query Definition requester, or an empty Query Definition requester is displayed.

2. Create a new definition, or edit the displayed one, as required. To remove a definition from the requester in order to create a completely different one, click on Clear.

3. When you are satisfied with the query definition, click OK to run it.

See also Saving a Query at the end of this chapter.

# The Query Definition Requester

The requester has three parts (see below). At the top is the Title Definition, below this is the Query Command Panel, and at the bottom left you will see four buttons for selecting the destination for the output from the query.

Query Definition works by accepting your selections and typed input through four requesters linked to the Fields, Report, Filter and Order buttons next to the Query Command Panel. Superbase then searches the file or files and outputs any records that match your Filter values.



## Defining the Query Title, Date, and Page Number

Title Definition has three parts: Date check box, Text box, and Page Number check box. If you make any selections or enter any text, Superbase will print the results as a Title at the head of each page or screen of the query output, followed by a blank line. If you don't define a title at all, Superbase starts the query output at the top of the page.

If you click the Date check box, the Title will include the system date, as shown in the Date Format requester on the Set menu. The date is always aligned with the left margin.

To enter the text you want for the query output title, click in the box between the Date and Page Number check boxes, and start typing. The maximum length is 80 characters.

If you click the Page Number check box, the Title will include a page number. The page number starts at one and increases by one every time it is printed. The page number is always aligned with the right margin.

Field names are output as the default column headings. You can redefine column headings with your own alternatives, or suppress them entirely. See 'Field' parameter, later in this chapter.

## Query Command Panel

The four lines of the Query Command Panel do most of the work of the Query command. Each line defines one of the elements of a query: output fields,

report features, filter conditions and sorting order; each of the first three can be up to 512 characters long.

- Fields determines which fields are output and allows you to format the output. Query output is normally tabular, but you can specify a page orientation if you prefer. See Defining Query Output, later in this chapter.

- Report allows you to include other features such as totals and record counts. It also sets subtotal breaks. See Chapter 15 Reporting.

- Filter controls the selection of records from a file. As a multi-file filter, it allows you to set up relational links between files. By doing this, you can make the selection of records in one file dependent on records in another file. See Defining a Query Filter, later in this chapter.

- Order specifies the order in which records are sorted. You can sort a file on one field – this sorts records in the same way as an index – or you can use several fields to sort a file at different levels. See Defining a Sorting Order, later in this chapter.

You do not have to use all four lines. For example, to produce a sorted report that prints just the selected fields for all the records in the current file, you would only use the Order line to specify the fields to sort on.

## Selecting Query Output Destination

You select the destination for the query output by clicking on the appropriate radio button on the Query Definition Requester. Then you run the query by clicking OK.

**Screen**     Directs output to the Screen.

**Printer**     Directs output to the Printer.

**Disk**     This selection creates a text file on disk containing the query output. The output is formatted as if it were being printed and is suitable for loading into your word processor or desktop publishing system.

By including stylesheet tags as text items within quotes in the correct places in the Query Fields line, you can preprocess query output into a form suitable for some popular desktop publishing programs.

**SB File**     Outputs a new Superbase file. A complete database structure with data file, definition file, and index is created automatically. See Chapter 16 Reorganizing the Database.

# Defining Query Output

The Fields feature allows you to select the fields you wish to view or print in the query output.

You can also specify other output features.

- Column heading
- Column position
- Number of characters
- Blank lines
- Derived expressions
- Boldface, Underline and/or Italic style
- Conditional form feed
- Column heading suppression
- One field per line output
- Inclusion of selected field list

When you want to define the output of a query, follow these steps:

1. Click the Fields button.

2. Select the file from which you wish to choose fields.

3. Select the fields you want in the order you want them.

4. Include additional features in the line as you select fields.

5. Click OK to return to the main Query Definition requester.

If you leave the Fields box blank, the selected fields from the current file will be output.

## Selecting File and Fields

At the top of the requester is the name of an open file. The fields in the list box below belong to this file. You can select any open file, and then select fields from it for output in the query.

1. Click the file selector button to the right of the filename to cycle through all open files.

   ☐ If only one file is open the field display does not change.

2. Select fields in the required order.

   ☐ You may wish to specify output features before you select a field. See below.

If you choose fields from more than one file, you are defining a **Multi-File Relational Query**. You will have to specify a link between the files. See Chapter 17 Multi-File Applications.

When you select a series of fields one after another, Superbase uses commas to separate the field names. When the data is output a space is placed between fields.

• If you replace the commas with semicolons, the items will be output without any spaces in between.

## Defining Derived Columns

A derived column in the query output is one in which the data is generated from a calculation or a function, usually involving an existing field or fields, one or more arithmetic operators, and explicit values. Examples are:

| | |
|---|---|
| Multiplication: | Quantity.Order * Price.Stock |
| Fixed decimal places: | FIX(amount.Order/3,2) |
| Increase of 15%: | Price.Stock * 1.15 |
| Decrease of 8%: | Price.Stock * 0.92 |
| Future date: | Date.Order + 30 |
| Day of the week: | DAY$(date.Order) |
| Gross margin: | Sales.Inv - (Cost.Inv + Commission.Inv) |
| Concatenated fields: | Firstname.Clients+ " "+ Lastname.Clients |

To set up a derived column:

1. Click on a field or an operator on the requester or:

   □ Type a value into the Value box and press ENTER or:

   □ Type a function keyword directly into the Fields box.

2. Continue until your expression is complete.

3. Use the AS operator to set a column heading if required (see below).

4. If necessary, add further fields or expressions.

A derived column that produces a numeric result is output in the default numeric format, which is specified with the Number Format command on the Set menu. See also Defining Numeric Format, below.

A derived date or time column is output according to the format specified with Date Format on the Set menu.

You can specify your own number, date or time formats for derived columns with the STR$, DATE$ or TIME$ functions with appropriate format strings. See Chapter 9 Using Functions. The data in a derived text column is output at its actual length, so its length may vary. This means that if the following fields are aligned, they must be positioned by specifying a fixed column position for the next column.

## Defining Column Headings

When you specify a field as part of the query output, Superbase normally prints its name at the head of its column on each page. You can use the AS Heading button to define your own column heading text for both fields and derived columns.

1.  Select the field or specify the derived column expression.

2.  Click the AS Heading button.

3.  Type the text for the heading into the Value box.

    ◻  If you're using Query to create a new database, you must make sure that the headings for any derived columns are valid field names – legal characters only, not reserved words, and not more than 15 characters in length. See Output Destination, below.

4.  Press ENTER to copy the heading text into the Fields box.

Examples:

> Sales.Inv - (Cost.Inv + Commission.Inv) AS "Gross Margin"
> Firstname.Clients + " " + Lastname.Clients AS "Full Name"
> Code.Clients AS "Postcode"

**Note**  If you wish to remove the column headings, enter the keyword FIELD as the first item in the Fields box.

## Defining Column Position

If you don't specify column positions for the fields, Superbase will output the field data for each record in a line, with each field appearing right after the previous one. Automatic positioning is useful for simple field output, but some derived columns are of unpredictable length, and you may wish to control the format manually.

To define a fixed column position for a field or a derived column:

1.  Click the @ Position button.

2.  Type the column number into the Value box.

3.  Press ENTER to copy it into the Fields box.

4.  Select the field or create the derived column expression.

You can use this feature to print fields from a single record on more than one line. For example, if you specify the same column position for two fields in a row, the second field will appear on the following line. See also Inserting Lines, below.

Overlapping fields can cause problems. This usually occurs when you tell Superbase to print a field at a column position which has already been used by

the preceding field's trailing spaces. Superbase's response is to print the field at the correct column position on the next line (see & Length, below).

## Defining Field Length for Output

One solution to the problem of overlapping fields on a line is to limit the number of characters output for each field. You may also wish to do this when you only need to output the first few characters of a field.

1. Click the & Length button.

2. Type the number of characters to be output into the Value box.

3. Press ENTER to copy it into the Fields box.

4. Select the field or create the derived column expression.

## Defining Numeric Field Format for Output

When you are outputting numeric fields, you may also use & Length to specify the number of digits that will be displayed before and after the decimal point. To do this, follow the procedure above, but enter a decimal number into the Value box in step 2. For example:

&5.3 (Numfield.Order/3)

Here the number 5.3 ensures that the result of the calculation (Numfield.Order/3) will be displayed to three decimal places and with five digits before the decimal point.

## Inserting Lines

Query output defaults to a tabular format determined by field lengths. You can vary this by inserting the NEWLINE keyword into the Fields line. This instructs Superbase to output the next item on a new line. Before or after any field or derived column expression:

1. Click the LINE button.

2. Repeat if more than one new line is required.

## Page Oriented Output

By combining multiple NEWLINE keywords and the Column Position feature, you can produce page oriented output suitable for printing on pre-printed stationery. The EJECT keyword can be used to move to a new page (see below). You may also wish to suppress column heading printing with the FIELD keyword (see below).

## Printing Long Text Fields

If you want to print text fields that are longer than can fit in a normal size column, you should follow a special procedure.

1. Place either a NEWLINE keyword or an @1 positioning parameter in the Fields line.

2. Type the function keyword TRIM$.

3. Place the name of the long text field in parentheses:

   TRIM$(Comments.CLIENTS)

Superbase will strip trailing spaces off the field and print it across the page, using wordwrap if required.

If you wish to print the field at its full length as given in the file definition, omit the TRIM$ function.

## Additional Format Parameters

There are a number of other parameters that can be used to specify the format for Query output. These are not shown in the Query Fields requester and must be typed directly into the Fields box with spaces or semicolons separating them from other words in the line.

The following table lists all the output format parameters that are not shown in the Query Fields requester:

| Parameter | Purpose |
|---|---|
| ALL | Includes the selected fields list from the current file in the output |
| FIELD | Suppresses column headings |
| DOWN | Outputs each item in an expression list on a new line |
| BF [ON] | Sets text to boldface |
| UL [ON] | Underlines text |
| IT [ON] | Sets text to italic |
| BF/UL/IT OFF | Turns selected text style attribute off |
| ATTROFF | Turns all text style attributes off |
| EJECT nn | Outputs data on a new page if there are less than nn lines left at the bottom of the current page |

### ALL
The ALL keyword causes the current selected fields list to be included in the output. You can add fields and expressions to the Fields box if you wish.

### FIELD
If you want to suppress the headings for all the fields in the command line, place the FIELD parameter at the beginning of the line. For example:

   FIELD Firstname.Clients, Lastname.Clients, Country.Clients

If you place FIELD anywhere else in the command line, it will only suppress the headings for the fields that follow it.

**DOWN**

DOWN must be placed at the beginning of the Fields line, before the field names. It ensures that the data for each field is output on a new line. For example:

FIELDS DOWN Lastname.Clients, City.Clients

would output data like this:

Arnauld
Paris

DOWN presents data like Record View, with leading field names. You can combine DOWN and FIELD , so that the data for each field is output on a separate line but without the field name, for example:

FIELD DOWN Lastname.Clients, City.Clients

**STYLEPARAMETERS**

BF, UL and IT set the text style for output.
BF sets boldface
UL sets underline
IT sets italic

Each of these can be set by ON (which is optional) and unset by OFF. ATTR OFF unsets all the text style attributes.

For example:

UL IT Lastname.Clients, UL OFF City.Clients ATTR OFF

The Lastname field will be printed or displayed on screen underlined, City will be italicized.

**EJECT**

You can use EJECT to ensure that all the data for a single record is output on the same page. The number following EJECT should be the number of lines occupied by the record data. Thus, if you were using two lines to output the data from each record, the Fields command line would look like this

Firstname.Clients, Lastname.Clients, NEWLINE City.Clients, Country.Clients EJECT 2

Superbase will then display or print a new page whenever less than two lines are left on the current page.

Entering EJECT on its own or EJECT 0 causes Superbase to output a new page for each record.

## The ON File Button

The ON File button is used when more than one file is involved in the query. Its function is described in Chapter 17 Multi-File Applications.

# Defining Report Features

The Report button call up a requester which allows you to include in your query definition various features commonly found in report generators. See Chapter 15 Reporting for details.

# Defining a Query Filter

The Filter button calls up a requester very similar to the Filter requesters used in the Print File function on the Project menu and in the browsing controls.

If you don't want to restrict the records selected for the query output, leave the Filter box blank. Data from all the records in the file will be output.

When you want to define a filter to select records for output, follow this procedure:

1.  Click the Filter button.

2.  Click the file selector button next to the filename to select the file from which you wish to choose fields.

3.  Create a Filter command line. See Chapter 10 Retrieving Data Interactively and Chapter 9 Using Functions.

4.  Click OK to return to the main Query Definition requester.

The Query Filter works exactly like other filters, except that you may cycle through the open files by clicking the file selector button. You click a field name, then an operator, then enter a value into the Value box and press ENTER . Use AND and OR to build up complex filters. Functions should be typed directly into the Filter box. See Chapter 9 Using Functions.

**Note**     If you choose fields from more than one file, you are defining a **Multi-File Relational Query**. You will have to specify a link between the files. See Chapter 17 Multi-File Applications.

# Defining a Sorting Order

The Order button allows you to choose a sorted order for the query output. You can select fields from any file, specifing either ascending or descending sorting order for each.

**Note**   You will need to have an area of free memory before you use Query Order.

If you don't specify any fields to order on, Superbase will present the records in the order in which it finds them on the disk. Most probably, they will not be in any alphabetic or indexed order. As this is the fastest way of retrieving information from the disk, you may want to use it if the order of presentation does not matter. If you're creating a new database you should probably avoid specifying an order, to speed up the processing.

However, in the majority of cases, you'll want to see the records ordered, especially when:

• Indexed order, such as date order, is an important part of the file. Also, defining a sort order on a single indexed field, allows Superbase to make use of the index and speeds up the Query.

• You have set up subtotals on the Report line.

If you do want to add order features to the query, follow this procedure:

1. Click the Order button.

2. If necessary, click the file selector button next to the filename to select the file from which you wish to choose fields.

**Note**   If you choose fields from more than one file, you are defining a **Multi-File Relational Query**. You will have to specify a link between the files. See Chapter 17 Multi-File Applications.

3. Select a field on which you wish to sort.

4. Click either the Ascending or the Descending button.

5. Repeat steps 2 to 4 for all the fields on which you wish to sort.

6. Click OK to return to the main Query Definition requester.

## Setting the Number of Characters on which to Sort

The default length for sorting is 15 characters per field. Superbase gives equal weighting to upper case, lower case and accented instances of a character. If you want a field to be sorted on more or fewer than 15 characters, use the & Length parameter; before selecting the field name:

1. Click the & Length button.

2. Type a number for the length and press ENTER. For example, to ensure that the Lastname field data was sorted on the first 20 characters, you would enter:

&20 Lastname

### Single and Multiple File Sorting

Outputting records from a single file is comparatively easy, and Superbase performs the operation without affecting the file itself. If you specified an order using just a single index, Superbase outputs the data as it finds it. But if your Order request includes several fields, you may have to wait a while for Superbase to create a temporary file prior to producing the query output.

However, if you are using more than one file, Superbase has to do quite a lot more work. Although the amount of work varies depending on the precise Order line you set up, essentially it means that Superbase can't tell what the final order is to be until it has finished reading all the files.

### Sorting on a Multi-field Index

If you often need to sort on more than one field, you may prefer to create a composite multi-field index, which Superbase will update automatically, allowing you to select it for sorting in Query Order and obtain results quickly. Such an index may be based on a Virtual field set up in the File Definition. See Chapter 3 Defining New Files and Chapter 4 Creating Indexes.

# Running a Query

To run a query, click OK on the Query Definition requester. The default output destination is the screen. If the Set Paging menu command is checked, the output will pause when the screen is full; press the space bar to continue.

# Saving a Query

Queries are re-usable. You can store the query instructions (except for the output destination) in a disk file, and open it when required.

1. Select the Process Query Save As command.

2. Select or enter a name for the query.

3. Click OK.

Superbase adds the extension SBQ and stores the query file. When you re-open it, any files that are not already open are opened automatically.

# 15 REPORTING

In Superbase, Reporting means more than simply listing data. If you just want to list your data without analyzing it at all, you may use the Print option in the Project menu.

At the other extreme, if you want fully formatted output, including multi-line headings and footings, conditional paging, multi-level sorting and grouping, and an extended range of statistical functions such as Standard Deviation and Variance, you should use the Form Designer Report Generator to create a Superbase DML report program. A report program may be opened and run directly from the database menus, or you may prefer to edit it with the DML Program Editor in order to fine tune certain aspects. For example, you could accumulate subtotal data in an array and perform a percentage analysis of all subtotals at the end of the report. In a fully programmed environment, you can also chain to a report program from a menu selection in a set of user-defined pull-down menus.

Between those two extremes lies the Process Query command, which provides some of the flexibility of a report generator while remaining easy to use. Chapter 14 Designing and Using Queries described all the features of a Query Definition except the Report line, and that is the subject of this chapter.

## Query Report Definition

If you want simply to output each relevant record from the file in the specified format, leave this line blank. Enter a Report definition if you want to include in your query certain features commonly found in reports, such as subtotals or counts of groups of records.

To enter a new Report definition, or to edit an existing one, click on the Report button in the Query Definition requester. A Query Report requester is displayed, allowing you to select field names and commands from which to build up your definition. A Report definition can be up to 512 characters long.

You can define a Query Report line to include keywords which specify a range of useful functions:

| Function | Keyword |
|---|---|
| Global report | REPORT |
| Report totals | SUM |
| Group subtotals | SUM |
| Group break fields | GROUP |
| Averages | MEAN |
| Record counts | COUNT |
| Summarization | SUMMARIZE |

If you do not want any of these features in your query, leave the Report line blank. If you do want some report features in your query output, follow this procedure:

1.  In the Query Report requester, first select the file from which you wish to choose fields for reporting features such as totaling. Initially, the name of the current file is displayed in the top box of the requester. If that is not the file you want, click on the upward arrow just to the right of the file name box. Each time you click on the arrow, the file name changes, with the names of all the available files being displayed in turn. Keep clicking until the name of the file you want is displayed.

2.  Click on one of the keyword buttons to the right of the panel containing the field names. See explanations below.

3.  If appropriate, select a field.

4.  Repeat steps 2 to 4 until you have defined all the reporting features you require.

5.  Click OK to return to the main Query Definition requester.

Before running the query, you will probably want to define other elements of the query, such as fields to be output, filter conditions, and sorting order. See Chapter 14 Defining and Using Queries.

## Defining Global Report Functions

You can define reporting features either for a subgroup of records or for the report as a whole. Global report functions are defined using the keyword REPORT .

1.  Click the Report button on the Query Report requester. This copies the REPORT keyword into the Report line.

2.  Click the Count button if you want a report on the number of records printed in the report. This copies the COUNT keyword into the Report line.

3.  Click the Sum button if you want to obtain totals for a numeric field. This copies the SUM keyword into the Report line.

4.  Select the numeric fields for which you want to obtain totals.

5.  Click the Mean button if you want to obtain averages for a numeric field. This copies the MEAN keyword into the Report line.

6.  Select the numeric fields for which you want to obtain averages.

7.  Click the Summarize button if you want Superbase to omit all output specified in the Query Fields line. This copies the SUMMARIZE keyword into the Report line.

Your Report line could look like this:

> REPORT COUNT SUM amount.Orders MEAN amount.Orders
> SUMMARIZE

This would produce summary figures for the total and average of the
**amount.Orders** field, and a count of the number of orders processed.

## The Concept of Record Groups

Before we look at the procedure for defining group functions, it is important
that you understand the concept of grouping on which the subtotal group
features are based.

A report consists of a series of records, which may be divided into groups
according to the sorting order you set. For example, in a file such as our
demonstration **Clients** file, all the records for each country may be printed
together, and all the records for each city may be printed together, by
specifying a Query Order line like this:

> Country.Clients ASCENDING City.Clients ASCENDING

In this example, the **City** groups would appear within the **Country** groups:

| Customer | Country | City |
|----------|---------|------|
| Customer 1 | England | London |
| Customer 2 | England | London |
| Customer 3 | England | London |
| Customer 4 | England | York |
| Customer 5 | England | York |
| Customer 6 | West Germany | Bonn |
| Customer 7 | West Germany | Bonn |
| Customer 8 | West Germany | Heidelberg |

A Record Group is defined by a change in the content of a field on which the
output is sorted. The change from Bonn to Heidelberg defines a **City** group,
and the change from England to West Germany defines a **Country** group.
Note that the change in the **Country** group necessarily implies a change in the
**City** group: West Germany terminates both York and England.

In this example, we would say that the file was Ordered or Sorted **by City
within Country**. As the file also has numeric or financial data, it would be
useful to generate a subtotal for each Country and City group. Averages and
record counts for each group might also be required.

## Defining Record Group Functions

Functions for Record Groups are defined with the GROUP keyword.

1.  In the Query Report requester, first specify any Global Report Functions (see above).

2.  Click the Group button on the Query Report requester. This copies the GROUP keyword into the Report line.

3.  Select a field that corresponds to one of the fields in the Query Order line. The field name is copied into the Report line.

    ☐ If you select a field for grouping that is not being used for sorting, your query output will not group all records with the same data in that field together.

    ☐ Usually the Group field is a text field, but if your report groups its records by date or number then you could obtain meaningful subtotals from a date or numeric field. The only fields you should not use as Group fields are those that change their values from record to record, such as most financial items.

4.  Click the Count button if you want a report on the number of records printed in the group. This copies the COUNT keyword into the Report line.

5.  Click the Sum button if you want to obtain subtotals for a numeric field every time the data in the group field changes. This copies the SUM keyword into the Report line.

6.  Select the fields for which you want to obtain totals in this group.

    ☐ If you have several levels of subtotals, you must repeat the SUM keyword and field name within each group for which you want to print subtotals.

7.  Click the Mean button if you want to obtain averages for a numeric field every time the data in the group field changes. This copies the MEAN keyword into the Report line.

8.  Select the fields for which you want to obtain averages in this group.

9.  Repeat steps 3 through 9 for each field on which you wish to group.

10. Click OK to return to the main Query Definition requester.

## Reporting on Derived Columns

If you want to apply report features to derived columns, you can. However, you must refer to the derived column expression by the name you gave it with the AS Heading feature in the Query Fields requester.

For example, the Query Report Command Line for subtotaling a derived column where the original expression was Trading_balance.Clients * 0.15 and the AS Heading text was '15%' would be:

GROUP Country.Clients SUM "15%"

You would enter '15%' into the Value box in the usual way.

- You must ensure that each item of AS Heading text is unique, or Superbase may confuse one expression with another.

### Overflow in Subtotals or Totals

When you request totals for a numeric field, you may sometimes create a format overflow. Suppose you had a numeric field with a format '999.00', defined at this length because the field itself can never exceed 999 in value. If the total were to run into four figures, as it would if even just 10 records had the value of 100 in this example, the format would not be able to handle it, and Superbase would print or display number signs:

###.##

To prevent this happening, you should adjust the field format in the File Definition. Select Project Modify File, select the field to be adjusted, and click the Numeric button. Increase the length of the integer part of the format, and exit from the Number format requester, then from the File Definition requester. There is no need to save the file definition at this point unless you wish to make the change permanent.

Now Superbase will use the expanded numeric format for all the numbers in the column, as well as subtotals and totals.

### Totals for Non-Printing Fields

You may request totals for fields that do not appear in the Query Fields Command Line. Superbase prints such totals without a column heading to the right of any totals for fields that are being printed.

## Report Example

This example illustrates the Report features we have mentioned. It shows you how to create a report on the data in the demonstration file **Clients**.

Let's set out our objectives in creating this report.

First, we want to sort customers into groups according to geographical area. The sorting order will be country then city; that is, the highest level of grouping will be by country, and within each country customers will be sorted according to their city.

Second, the report will show the totals for the two numeric fields in each **City** and **Country** Field Group. It will also show record counts for each group.

Third, the report will show totals and records for the file as a whole.

How do you create a report which satisfies these requirements? The following sections take you through the procedure step by step.

Before you begin, use Query Fields to select some fields from the **Clients** file for output. Select at least **Country** and **City**, in that order. You can leave the Query Filter line blank, but select Query Order and specify a sorting order of

Country.Clients ASCENDING City.Clients ASCENDING

Now you are ready to add report features.

## Global Report Totals, Averages, and Record Counts

The first step is to define the requirements for the report as a whole. Select the Query Report requester, and ensure that the **Clients** file is selected.

1. Click Report. This tells Superbase that the selections you are going to make will apply to the report as a whole.

2. Click Sum. This indicates that the fields you are going to select are to be totalled for the report as a whole.

3. Click the **Trading_Balance** field in the Fields List box. Superbase copies the name into the Report line after the keywords. Repeat this action for the **Cash_receipts** field. The Report line so far will be:

   REPORT SUM Trading_balance.Clients, Cash_receipts.Clients

4. Record counting. This feature is either on or off. As we want the records to be counted and the total printed at the end of the report, we must use this option. Click Count. Superbase copies the keyword into the Report line:

   REPORT SUM Trading_balance.Clients, Cash_receipts.Clients
   COUNT

## Adding Group Totals, Averages, and Record Counts

Now we come to specifying report features for the different groups within the report.

1. Click Group, and then on the **Country** field name. The Report Line has this added to it:

   GROUP Country.Clients

Superbase applies the selections you make next to the named group only.

2. Click Sum. This indicates that the fields you are going to select are to be totaled for the **Country** group.

3. Click the **Trading_balance** field in the Fields List box. Superbase copies the name into the Report line after the SUM keyword. Repeat this action for the **Cash_receipts** field. The Report line is extended to show:

   GROUP Country.Clients SUM Trading_balance.Clients,
   Cash_receipts.Clients

4.  As for the report as a whole, the record counting function is either on or off. By definition it counts only the number of records in the group. Click Count, and Superbase copies the keyword into the Main Box:

GROUP Country.Clients SUM Trading_balance.Clients,
Cash_receipts.Clients COUNT

Your next step is define a **City** group. Do this in the same way as **Country**. The final Report line should read:

REPORT SUM Trading_balance.Clients, Cash_receipts.Clients
COUNT GROUP Country.Clients SUM Trading_balance.Clients,
Cash_receipts.Clients COUNT GROUP City.Clients SUM
Trading_balance.Clients, Cash_receipts.Clients COUNT

It looks lengthy, but when you read it through it's quite easy to understand what's going on.

When you have obtained the Report line above, click OK to return to the main Query Definition requester.

## Example Output

If you now click on the OK button in the Query Definition requester, the first page of our example report will look like this:



| Country | City | Balance | Receipts |
|---------|------|---------|----------|
| Australia | Freemantle | £3,039.86 | £740.92 |
| Australia | Freemantle | £1,077.00 | £0.00 |
| | | £4,116.86 | £740.92 |
| Group Count 2 | Freemantle | | |
| Australia | Sydney | £7,832.95 | £4,012.91 |
| Australia | Sydney | £8,500.00 | £3,000.00 |
| Australia | Sydney | £1,800.00 | £560.00 |
| | | £18,132.95 | £7,572.91 |
| Group Count 3 | Sydney | | |
| | | £22,249.81 | £8,313.83 |
| Group Count 5 | Australia | | |

## Avoiding Meaningless Command Lines

The order of your selections when building the Query Report line can be significant. By clicking at random on operators and field names, you can produce a meaningless Report line. This is because you are in effect **programming** Superbase to obey your instructions, and along with this extra power goes an increase in responsibility.

You need not be concerned, because you cannot harm your data, but it is best to learn through practice how to avoid 'garbage out' by eliminating 'garbage in.' A line like this cannot be interpreted meaningfully:

SUM MEAN Lastname, COUNT Cash_receipts REPORT GROUP

Superbase will either show an error message if there's an error in the line (such as trying to sum a text field), or print meaningless data.

# 16 REORGANIZING THE DATABASE

A database can be reorganized in various ways for different reasons. This chapter covers:

- **Restructuring a database file.** From time to time, you will want to change more than the characteristics of individual fields or the layout of a Page View. With experience, you will see that the fields in your database need to appear in a different order, perhaps with extra fields added and others deleted altogether.

- **Reorganizing a database.** If you do much editing of a file, adding and deleting records, you will also need to reorganize it from time to time in order to obtain maximum efficiency. Database reorganization can also enable you to recover as much data as possible from a damaged database.

- **Splitting a database file.** Superbase provides facilities for using existing data and definition files as the basis of new database files.

- **Merging database files.** Superbase provides facilities for merging two databases together.

## Restructuring a Database File

You restructure a database by reading all the records you need from one or more existing database files and creating with them a new database file. Use the Process Query Definition requester to specify the fields and files to be read, the order and formats of the fields in the new records, and the name of the new database file to be created.

Existing field definitions become the basis of the new SBD file. The new file is automatically indexed on the first indexed field encountered in the Query Fields line, unless no indexed fields are being output, in which case the index is built on the first field.

1. Select Process Query Edit if you want to build up the required Query Definition from scratch, or Process Query Open if you already have the required Query Definition or one close to it.

2. Use the Query Fields requester to define the records for the new database. See Chapter 14, Defining and Using Queries, and Chapter 17, Multi-file Applications. You may want to define a Filter to include or exclude some records.

   □ Do not define a Query Report line.

3. Be sure to click on the File button, to specify that the destination for the output of the query is a file.

4.   Click on OK.

5.   Supply the name for the new database file and click on OK.

## Defining the Query Fields line

Careful use of the facilities provided can greatly enhance the usefulness of the feature. Here we list some points which might be overlooked.

*   Fields can appear in any order. Often, the need to restructure arises because you have added a field to the file definition, which means it appears at the end of the list of fields. A common use of this feature is simply to rearrange the file definition.

*   Certain formatting keywords and parameters are ignored by Query when outputting to a File. These include:

    **ON File**
    **EJECT**
    **NEWLINE**
    **@ Position**

*   Derived columns. The results of expressions in the Query Fields line will be output as field data of the appropriate type. The name defined with the AS Heading feature will be the field name, so ensure that it has no more than 15 characters, that it obeys the rules for field names, and that you do not use the same name more than once. See Chapter 3 Defining Files.

*   Non-field data. If you include literal text or numeric values in the Query Fields line and give them Heading names, they will be defined as new fields with data in them.

*   A Calculated field will lose its formula unless all the component fields for the formula have preceded it in the Query Fields line. You may force a calculated field to lose its formula by enclosing it in parentheses.

*   Page View. Superbase does not recreate each field's location co-ordinates. You must reposition the fields and store the new co-ordinates with the Project Save command.

## Reindexing

Restructuring a database file creates only one index, as explained above. This considerably speeds up the output process, but it does mean that you must rebuild all other indexes with the Project New Index command.

## Multi-file Operations

Query output from more than one file can be used to create a new database file. A relational Filter line must be defined. See Chapter 17 Multi-File Applications.

This command should not be used to merge two files. See Merging Database Files at the end of this chapter.

# Reorganizing a Database

The need to reorganize arises periodically as a result of the way Superbase maintains its internal file structure. Reorganization compresses the data blocks, eliminating all deleted items.

When you delete records from a file using the Process Remove option, Superbase does not actually remove them from the disk. Instead it simply marks them as deleted records and thereafter re-uses them in file operations. (You can check this by noting the file size with the Status file option, before and after selecting Process Remove.)

Superbase will store new records in the space which was occupied by the deleted records. But as far as other programs are concerned, the records are still there and the space is unavailable. If you have removed a large number of records – say, over a long period of time – you may be left with a significant amount of wasted disk space. Removing a large number of records from one file may also mean that Superbase takes longer to read records from the file.

Reorganize works with the current file and builds a new file using the same name, the same file definition, and the same data, but without the deleted records. It allows you to reclaim unused disk space.

When you need to reorganize a file, follow this procedure:

1.  Open the file to be reorganized.

2.  Select Process Reorganize.

3.  Enter a name for the new file.

    □  A pathname is allowed.

    □  You can enter the same name as the original if you wish, but this will cause it to be overwritten.

4.  Click OK.

Superbase builds up a new file from the source file, then recreates all the indexes. Fields marked DEL are removed from the file definition.

After the reorganization, the new file is opened and made the current file.

•   This option should only be used periodically and not on a regular basis. Unless you have deleted a large number of records, there is only a small advantage to be gained by reorganizing a file. Before reorganizing a file, you should always make a backup copy of it.

- Check that there is enough space on the destination disk to accommodate the reorganized file. As a rule of thumb, this should be at least as much as the space occupied by the current file.

- You may want to reorganize files which make use of the function to generate unique serial numbers. In this case, you should be aware that the numbers in the new file continue from the highest number in the old file.

- If you reorganize a file using its original name, Superbase first makes a backup copy of it under the filename **DBXREORG**. The SBF, SBD, and all index files bear this root name during the process. **DBXREORG** is deleted after the reorganization is completed. This means you can recover your data if for any reason the operation is interrupted (for example, by a power cut): simply rename as your original file all the files with the name **DBXREORG**.

- File names in quotation marks that occur within calculation or constant formulas are not changed when you restructure a file and may therefore need to be updated. For example, if the original file contains the formula SER("Clients"), you will have to replace 'Clients' with the name of the new file.

## Recovering Data by Reorganization

Database work is very disk intensive, and thus more susceptible to disk errors than spreadsheet or word processing work. If Superbase starts to display error messages which indicate the presence of damaged records in the file, you should use Reorganize to reconstruct it immediately.

Superbase reads the database without using an index, which enables it to regenerate indexes from the actual data. If any data records are damaged, they are omitted from the new file. If this occurs, Superbase shows the following information message after completing the reorganization.

New file has different record count

There may also be circumstances in which Reorganize can be used to recover data which could not be accessed before. In this case, Reorganize rewrites the file header so that the records are included in the new file. Then it issues another message (as well as the one above):

Source file size incorrect in header

No user action is required following these messages.

## Creating an Empty Duplicate File

Applications often require the creation of empty files with definitions based on the old one. You can achieve this using the Empty Copy option in the Project New menu. See Chapter 3 Defining New Files for more information.

## Splitting a Database File

You may need to generate two separate files of the same structure from a single database file. This is achieved with the Split command on the Process menu.

1. Open the file that you wish to split.

2. Select the Process Split command.

3. Define a Filter Command Line with values that will select the set of records you wish to go into the TRUE destination file – where all records meet the filter criteria.

   □ The records that do not match will go into the FALSE destination file.

4. Click OK.

5. Enter filenames for the Superbase TRUE and FALSE files that are to receive the extracted data and click OK.

## Merging Database Files

Applications often call for two database files to be merged together. Because Superbase's file structure is more sophisticated than those of traditional 'flat-file' databases, it is impossible simply to append one file to another and reindex the resulting file. Superbase requires that new data be inserted into an existing file and index structure. The benefit of this approach is that all reindexing is done automatically.

For any pair of files to be merged:

1. Decide which file is to act as the 'host' file. If there are no other factors, make the larger file the host file.

2. Open this as the current file.

3. Import the other file into the host file, using the Superbase Merge option.

• If a selected fields list is active in the source file, Superbase will use it. This allows you to select fields in any order for merging into a host file.

See Chapter 21 Import for more information.

# 17 MULTI-FILE APPLICATIONS

Superbase includes a number of features which may be combined to create practical multi-file applications. Files may be linked for data entry and in forms created with the Form Designer. Relational queries and updates provide ways of joining files for processing operations.

In this chapter we discuss the principles which you should follow and the program features you should be aware of when you are considering whether your own application needs to be multi-file.

If you are going to set up a multi-file application, the descriptions of file types, the principles of linking files, and the rules for defining queries should enable you to make a start on analyzing your requirements. Be prepared to spend some time working with prototype files before committing yourself to the real thing.

A multi-file application is one in which the user's requirements for data entry, retrieval, and reporting cannot be satisfied by a single file structure. The real-world model for a single file or flat-file database application is the card index box, in which each card represents a record, and each record is a collection of unique fields. It is relatively easy to set up such a system, and reporting on a single file is largely a matter of selecting fields and records for output.

A file of book references, for example, can be listed and grouped in many different ways – by author, title, publisher, and year of publication. Provided you are satisfied with the results, a single file is appropriate. However, a simple decision to store more details about an author than just his or her name changes the nature of the application. You would not want to waste disk space and retrieval time by storing the author's place of birth with every book. Instead you would keep all the details about an author in one place, and all the details about books in another, and link the two files together: a multi-file system.

Immediately, you must start to explore questions such as, how should the two files be linked? Should the links be permanent or temporary? How can I be sure that the data is properly cross-referenced?

Superbase provides ways of answering all these questions. Forms, cross-file lookup, relational queries, and global multi-file updating commands can be combined to produce effective and flexible applications of many kinds.

# Files for Multi-File Applications

The relationships between the files in an application and the structure of the files themselves are determined by the nature of the data.

Files themselves can be usefully categorized into three types: those containing static or 'tombstone' data, such as author's date of birth; dynamic or transactional data, such as a publication by an author; and tables of data such as a list of topics or keywords for use in categorizing other data. All these files are set up in the same way as normal Superbase files. The division into three types is purely logical.

There are also three important types of relationship: one to one, one to many, and many to many. A look at the file types will help us to clarify the meaning of one to many and one to one relationships.

## Tombstone Data

This type of file is used for storing details about a person or thing (including legal entities such as companies). Names and addresses, customer account details, product details, details of services offered, personnel details – these are typical examples of tombstone data found in the commercial area. Comparable examples can be found in education and other domains, ranging from student or patient records to laboratory experiments and libraries.

Tombstone data does not change frequently, and only one instance of it needs to be stored. In multi-file applications, these files serve as 'master' files, with a unique reference field (see Defining Common Reference Fields, below), usually a code of some kind, which is used to link each record with associated transaction records in other files in one to many relationships.

During data entry of transaction records, details from the tombstone record may be looked up to ensure that the same data is entered correctly into every associated transaction record.

One important type of field often found in tombstone records is the balance field. This is not necessarily a money amount: inventory counts are also balances. The important point is that the balance is held in the master record, and not in the transaction record, although data from the latter is often used to update the former.

## Transactional Data

This type of file provides the other end of the one to many relationship. Here the records consist of 'events,' although thinking of them as such may sometimes seem rather unfamiliar. Orders, stock movements, client billings, publications, exam results, book lendings, and laboratory instrument readings are examples of transactional data which is inherently linked to a single tombstone record. Each such record will include a common reference field

which links it to its unique master record. Several transaction records will contain the same reference, so that, for example, all an author's books can be linked to the same author record.

## Table Data

There is some overlap between the concepts of the tombstone data and table data, in that table data is also static and only one instance of it is stored. A file of country names should not need to change, and each country only occurs once. Table files provide a means of ensuring that data to be used for subsequent searching is always entered in a standard form. Although tables can be quite large, they often contain fairly small lists of codes or words which occur repeatedly in other records. Typical tables are lists of countries or states, standard codes used for categorizing businesses or individuals or products, and lists of key words used for searching transactional files such as bibliographies.

The relationship between a table record and a record in a tombstone or a transaction file is one to one. There should be only one occurrence of the table data per tombstone or transaction record; if there is more than one, you are breaking an important rule which forbids the repetition of fields in a record structure (see next section).

Tables are heavily used in data entry. For example, when the user wants to enter a country name into an address record, the system can display a list of all the allowable countries. The user selects the required country name, which is copied in automatically, thus ensuring that it will be spelled the same way in every record in which it occurs. (You can also arrange for the country name not to be copied, thus reducing storage requirements.)

## Avoiding Repeating Fields in a File Definition

This section illustrates the rule that a record should not contain repeating fields. One of the most common errors when setting up databases is the repetition of identical field structures in a file definition. In a database of authors and books, it would be tempting to define a single file with a record for each author, in which you store details of the author such as his or her name, plus fields to hold the details of, say, 10 or 20 books. The book fields, you might think, could be given different names, such as Title1, Title2, etc. It seems so simple, yet it would greatly reduce the usefulness of your system.

If your application requires a one to many relationship like this, and as we saw in the previous sections, very many applications do, then you must set up two files, one for the 'ones' and one for the 'many': **Authors** and **Books**.

By defining a common reference field for the two files, such as **Author_code**, you can achieve a supple and dynamic application which will ensure accurate data entry and efficient data retrieval.

In short, never define groups of repeating fields in a file definition if you wish to make the best use of Superbase's relational features, which are designed to help you achieve the best possible data management system.

# Data Integrity

A library system will be useless if you cannot be certain that the author code is correctly entered into every publication record. Therefore, the integrity of the data to be used for linking files must be assured. This is achieved by predefining a capability for linking through common reference fields, and by ensuring that reference field data is validated when it is entered, both in its own terms and by automatically checking it against existing data.

## Defining Common Reference Fields

Every pair of files between which there is to be a relationship must have a reference field in common. This is a field in each file that holds the same data. The common reference field is used as the link between the pair of files during data entry and when relational queries and updates are defined. These fields should be designated as indexed fields. At least one of the indexes should only allow unique keys.

Examples of common reference fields are account codes, part numbers, identity numbers, etc. Common reference fields are usually text fields, and are kept as short as possible to speed up index updating and sorting during queries. Pairs of common reference fields do not have to have the same name in both files.

## Common Reference Field Validation

At the data entry stage, what goes into the common reference fields must be carefully controlled. The aim is to ensure that, for example, an account code is always entered in the same case in all files in which it is used. This is done in three ways:

1.  By setting the text field attribute for each of the common reference fields to perform the same type of automatic case conversion, to either upper or lower case.

2.  By where possible using a calculation to generate the contents of the common reference field automatically. For example, the serial number function SER can be used to create a unique number for a new record, which can be combined with part of a text field such as a person's name.

3.  By attaching a validation formula to check the pattern of what is entered, its length, or the range of acceptable values. This may only be necessary in certain applications.

### Common Reference Field Cross-File Lookup

A further control on data integrity is via the LOOKUP function. This enables
the system to check what is entered into the common reference field in one
file against what already exists in the common reference field in the other file,
which may be either a tombstone or a table file. For example, you would
always want to check that a product was in stock before accepting an order for
it.

The LOOKUP function is attached to the common reference field in the file
definition as a validation formula. It uses the data you enter as an index key for
the common reference field in the other file, and attempts to read a record
from that file. For example,

> LOOKUP (stockcode.ORDERS,stockcode.STOCK)

would be the validation formula for the **Stockcode.Orders** field. When you
enter data into this field, Superbase uses it as an index key for the
**Stockcode.Stock** field, and attempts to read the corresponding record from the
Stock file. If the record is not found, the validation fails.

If the record is found, any fields which are cross-file references, in which data
in the record being entered is copied from the looked up file, are completed
automatically.

The LOOKUP function can be extended in two ways. You can add a user
defined help message, such as 'This Stock Code Is Not On File.'

Or, more usefully, you can add a REQUEST 20 command to the validation
formula. This command specifies a requester which displays selected data
from the file being looked up, in our example the **Stock** file. The user can then
select the correct record from a list, thus ensuring that data is never invalid.

See Chapter 18 Linking Files for full details of LOOKUP and REQUEST 20.

# Forms for Multi-File Applications

The underlying structure of a multi-file application resides in the file
definitions you set up using the concepts and techniques described above.
Forms provide many ways of extending and refining an application.

You can have many different forms for an application. In our Authors and
Books example, you could have one form for entering details of authors and
another for entering details of books. A third form could show a list of books
for an author, and yet another could show just a table of books. One form
could allow editing, another could be restricted to reading only.

Yet all these forms would use the same file structures, so when you added new
data or edited existing data the same LOOKUP functions would ensure the
integrity of your data whichever form you were using.

If you plan to set up a multi-file application, first define your file structures, then use the Form Designer to build the forms you need as you need them. The Form Designer has many features to appeal to the more advanced user, such as 'one-to-many' transaction handling, specialized calculation functions, and the ability to include DML command lines that can interface to subroutines in a resident DML program. However, the average user will be able to create an attractive data entry or browsing form in minutes.

## Summary

Here is a summary of points to bear in mind when creating a multi-file database system.

- Analyze your requirements to see whether you need a flat file or a multi-file system. If your data breaks down into tombstone and transactional data, you probably have a multi-file application.
- Decide whether you should keep any frequently occurring data in tables.
- Set up file definitions for all the files to be linked.
- Do not repeat fields within a file definition.
- Define common reference fields for each pair of files to be linked for data entry, Update, or Query.
- Use validation formulas to control the range and pattern of data entered into common reference fields.
- Set the same Text Field case conversion attributes for each pair of common reference fields.
- Use LOOKUP to cross-check data entered into one file against data already present in the other.
- Use REQUEST 20 to ensure that data entered into one file is copied from data already entered in the other.
- Define fields as virtual if you want to avoid repeating tombstone or table data in transaction records.
- Keep record and index key sizes as small as possible.
- Use as few indexes as possible.
- Create forms after having created and tested file definitions.

# Data Entry in a Multi-File Application

Data entry is critically important to a successful multi-file application. The integrity of the data must be maintained so that you can be confident that subsequent searching and reporting produce accurate results.

## Preparing for Data Entry

The order in which you prepare your files is important. All table file data must be entered first, so that other files can refer to it. If you are using a table of country names and while entering names and addresses you need to enter a country which is not in the table, you must stop and make an entry into the table before proceeding.

Likewise, all tombstone or master file data must be entered before you start entering transactional data.

## Using Lookups

When you enter data into a field which is linked to another file with LOOKUP, Superbase uses the characters you type as an index key for the other file. As explained above, there are a number of possible outcomes for the lookup operation.

If you typed in an exact key, Superbase finds the correct record in the other file. Any cross-file references in the record you are entering will be completed and the fields 'filled in' for you.

If the key you typed is not found, the validation fails; if you have specified an error message it is shown. You may click either Cancel or OK. OK puts the insertion point back in the field to try again. Cancel terminates data entry.

If you have included a REQUEST 20 extension to the LOOKUP validation formula, Superbase shows a list of data from the other file. The starting point of the list depends on what you typed. For example, typing 's' will display records from the nearest key value to 's' onwards. This technique allows you to position quickly to a record close to the one you want, then visually confirm the exact record required.

Pressing ENTER in a LOOKUP field without having typed a space or any other character passes through the field (unless it is a Required field, in which you must enter data). However, you will not be able to save the record without having performed a successful lookup.

## Entering Data into Multiple Files

Forms allow you to enter data into several files from a single screen.

Following a Record New command Superbase prepares to create a new record for each file associated with the form. It will only actually create a record in a file if you enter data into one or more fields for that file. So if you do not want to create a record in a certain file – for example, you might not want to allow anyone to add a new author on the same form you use to update the file of books – just designate all the fields from that file Read Only when you create the form.

This allows you to have any number of fields shown on a form for display or reference purposes, without having to be concerned about the security of the data.

This is also true when you edit existing data. If none of the fields for a file are changed, Superbase will not update that file.

The Form Designer also allows you to specify the order in which data is entered into fields.

## Transaction Blocks

One of the most powerful features offered by the Form Designer is the ability to define a group of fields as 'transactions.' Using this feature, you can create a form that manages a one-to-many relationship without any need for programming.

Each transaction consists of one or more fields defined as a group. The group is replicated in a special way that allows Superbase to create multiple records in the file at the 'many' end of the one-to- many relationship when you save the form.

Special functions may be added to the form to display running totals, post data from one file to another, etc.

You enter data into the fields in a transaction block in the same way as for ordinary data entry. Each transaction can use the LOOKUP and REQUEST 20 function to cross-check data, and may itself include references to fields in more than one file – although only one one-to-many relationship is allowed per form without extra programming.

## Cross-File References for Non-Form Data Entry

If you are not using a multi-file form for data entry, you may need to set up cross-file reference fields to copy data from looked up files. For example, when entering the details of a publication you would look up the author's record by means of an author code, and you would have set up the file to copy the author's name into the record you entering. The field **Name.Book** would be copied from **Name.Author**. **Name.Book** would in fact be a calculation formula of this type:

```
LOOKUP(Author_code.BOOK,Code.AUTHOR) ?
Name.AUTHOR : ""
```

The way you position cross-file reference fields in Page View can make a difference to the way data entry works. It is usually preferable that cross-file references are completed immediately after the lookup is performed. Since cross-file reference fields will always be Read Only, the insertion point passes over them into the next editable field. The recommended method is to place all cross-file reference fields that are completed following entry into a

LOOKUP field immediately after the LOOKUP field, in sequence from left to right and top to bottom. This will ensure a satisfactory display.

## Using Virtual Fields

Virtual fields can have a useful role in a multi-file application. If your customers come from a group of twelve countries, you might set up a table file **Country** to hold the country names, and copy the appropriate name into each customer record when you enter it. **Country.Customer** will therefore be a cross-file LOOKUP field. By making **Country.Customer** also a virtual field, you will not have to store the country name in the customer file, thus saving on disk space.

However, because the contents of **Country.Customer** must be looked up from the **Country** file every time a record is read from the **Customer** file, there is an inevitable effect on performance. The effect will be more or less noticeable depending on the speed of your computer. If the table file is small, Superbase may be able to hold all the data in its own buffers and minimize the overhead. You will have to decide for yourself whether you want to trade disk economy for speed, or not.

# Defining and Using Multi-File Queries

You should have read Chapter 14 Defining and Using Queries before attempting to define a multi-file query.

Superbase provides excellent facilities for linking files together in queries. You can specify fields for output from many different files, and set links between many pairs of files, subject only to the 512 character limit on the Query Fields and Filter command lines. In practice, you are unlikely to need to set links between more than four or five files in a single query.

This is a very powerful facility, and the user has to accept a certain responsibility for learning to understand the rules that Superbase's query logic obeys. The definition of efficient queries is therefore a skill to be learned, and a period of experimentation may be necessary before you are fully satisfied with the results.

## Defining a Multi-File Query Fields Line

To output fields from more than one file, you follow the same procedure as for single-file queries:

1. Open the files whose fields you wish to output.

2. Select the Query command on the Process menu.

3. Click the Fields button in the Query Definition requester.

4.  Initially, the List box shows the fields in the current file. Click the file selector button (the right arrow) to the right of the file name to select another file.

    ❑  You can cycle through all the open files by repeated clicking on the file selector button.

5.  Before selecting a field, specify any output formatting features you may require by clicking on their respective buttons; for example: @ Position, & Length, As Heading. See Chapter 14 Defining and Using Queries for more information on these features.

6.  Select the fields you want in the order you want them. To enter a field name in the Query Fields text box, double-click it or click once and press ENTER .

7.  When you have completed the Query Fields command line, click OK to return to the main Query Definition requester.

## ON File

Outputting data from more than one file raises a special problem of formatting. The ON File option is designed to solve it.

ON File causes specified field data to be displayed only when record data changes. Certain filter command lines (see Optimizing Queries, below) may cause Superbase to retrieve records in an order which effectively changes the record data on each iteration. This may result in the unexpected display of duplicated field data which ON File normally suppresses.

The best way to explain how ON File works is with an example. For this purpose, we will use one of the demonstration Query files, **Clibals1**.

Load **Clibals1** using the Query Open option. **Clibals1** automatically opens the two database files, **Clients** and **Trans**. Both files hold data relating to the same set of individuals. The **Trans** file contains details of the stocks transactions made by the individuals in the **Clients** file, using one record for each transactions. Note that the relationship between the two files is 'one to many': for each record in the **Clients** file, there may be several records in the **Trans** file.

**Clibals1** outputs data from both files, using the **Customer_ref** field as a link. This link is set up in the Query filter, which is explained later.

The Fields line in the **Clibals1** Query file specifies which fields are to be shown in the report:

```
&20@2Client.CLIENTS AS "Client" @24Trans_Date.TRANS AS "
Date",Transaction ref.TRANS AS "Ref",USD Value.TRANS AS
  "  $ Value"
```

When you click on OK in the Query requester, the screen output should look like this:

```
1-Sep-1989                    Client Trading Balances

Client                    Date        Ref        $ Value

Pierre Arnauld            Apr 02, 89  000005      945.95
Pierre Arnauld            May 03, 89  000025      425.68
Pierre Arnauld            May 05, 89  000030     -464.53
                                                 -------
                                                  907.10

Jean-pierre Broonen       Apr 02, 89  000001      425.68
Jean-pierre Broonen       May 03, 89  000018     2956.08
Jean-pierre Broonen       May 05, 89  000027     -295.61
                                                 -------
                                                 3086.15

William Cellers           Apr 02, 89  000002    22500.00
William Cellers           Apr 02, 89  000009     3547.30
William Cellers           May 03, 89  000020    19288.96
William Cellers           May 03, 89  000022     9909.23
William Cellers           May 05, 89  000029   -22312.50
William Cellers           May 05, 89  000034    -3527.03
William Cellers           May 05, 89  000035     3463.48
                                                --------
                                                32869.44
```

Too much data is repeated. What is needed is a way of telling Superbase which details to print for each file. ON File does exactly that.

The demonstration Query file **Clibals2** works with the same database files as **Clibals1**, but uses ON File to format its output. The only difference between these two Queries is in the Fields line:

> ON "CLIENTS"&20@2Client.CLIENTS AS "Client" ON
> "TRANS"@22Trans_Date.TRANS AS "Date",Transaction
> ref.TRANS AS "Ref",USD Value.TRANS AS "   $ Value"

If you now load **Clibals2** and click on OK, the screen output will be like this:

```
1-Sep-1989                    Client Trading Balances

Client                    Date        Ref        $ Value

Pierre Arnauld            Apr 02, 89  000005      945.95
                          May 03, 89  000025      425.68
                          May 05, 89  000030     -464.53
                                                 -------
                                                  907.10

Jean-pierre Broonen       Apr 02, 89  000001      425.68
                          May 03, 89  000018     2956.08
                          May 05, 89  000027     -295.61
                                                 -------
                                                 3086.15

William Cellers           Apr 02, 89  000002    22500.00
                          Apr 02, 89  000009     3547.30
                          May 03, 89  000020    19288.96
                          May 03, 89  000022     9909.23
                          May 05, 89  000029   -22312.50
                          May 05, 89  000034    -3527.03
                          May 05, 89  000035     3463.48
                                                --------
                                                32869.44
```

As you see, ON File ensures that the clients' names and addresses are not repeated. It instructs Superbase not to output fields from a file unless the current record has changed. You can think of ON as being short for 'on change of record.'

In this example, Superbase reads a record from the **Clients** file and then reads the records in the **Trans** file. Every time it finds a **Trans** record whose **Customer_ref** field matches the **Customer_ref** field in the **Clients** file, it outputs the data specified by the **Fields** line. Without the ON "Clients" command, it repeats the name and street from the **Clients** file for matching records in the **Trans** file. The effect of ON "Clients" is to suppress duplicate names and addresses.

The second instance of ON File – ON "Trans" – has a different effect. It cancels the first ON command and instructs Superbase to output data for all the matching records in the **Trans** file. If you left out ON "Trans" from the Fields line, only the first matching record for each client would be printed.

Superbase assumes that the ON command applies to all the field names to the right of it, irrespective of which file they belong to. This is why we need to include ON "Trans" in the Fields Line; it ensures that the first ON command only applies to the fields in the **Clients** file.

If we were not concerned about the order in which fields are output, we could dispense with the second ON command, as in the following Field line:

    Description.Trans, Total.Trans, Trans_date.Trans ON "Clients"
    Firstname.Clients, Lastname.Clients

## Entering ON File

To include this in the Fields command line, click on ON File in the Fields requester. Superbase will then copy ON into the Main Box together with the name of the file that is in the File Name Box.

Note that ON File only works when the data for each 'level' of the Query output is held in separate files.

## Defining a Multi-File Query Report Line

The same considerations apply to multi-file as to single-file reporting. See Chapter 15 Reporting for more information.

1. Select Process Query.

2. Click the Report button on the Query Definition requester.

3. Click the file selector button at the right of the file name to select the file from which you wish to choose fields.

4. Enter the report commands by clicking on one of the keyword buttons to the right of the fields list.

5. If appropriate, select a field.

6. Repeat steps 4 to 6 until you have defined all the reporting features you require.

7. Click OK to return to the main Query Definition requester.

## Defining a Multi-File Query Filter

A query is multi-file when fields from more than one file are named in the Query Fields or Filter command line. For a multi-file query to work, you must specify a link between the files. This is done by means of common reference fields (see above). Each file in a pair of linked files includes a reference field in which the data is the same as in the other file, such as an account code.

This link is known as a join. In Superbase, a join is defined in the Query Filter line as two fields joined by an EQUALS operator:

Customer_ref.Clients = Customer_ref.Trans

This is referred to in this section as a join expression.

• You can have joins between more than one pair of files.

• A file or field may appear more than once in the series of joins.

Superbase allows you to create a Query Filter in almost any form. However, query filter definition requires careful attention as certain rules must be observed.

1. Only the = operator can be used in a join expression.

2. Both fields must be indexed.

3. If you have more than one join in a filter, the join expressions must be related with the AND keyword:

Customer_ref.Clients = Customer_Ref.Trans AND
Stock_key.Trans = Reference.Stock

4. Do not use an OR keyword to relate join expressions.

5. Place all join expressions at the start of the line.

6. Every file reference in the line must be part of a chain. A line of the form A=B AND B=C is acceptable, but A=B AND C=D is not. A simple rule of thumb is that there should always be one less join than the number of files in the query.

**Note**     These rules must be followed. If they are not followed, you could end up with queries that produce repetitive and meaningless results, and take a very long time to do so.

• If you want to interrupt a query, click the Stop button or press CTRL+C .

## Optimizing Queries

To some extent, the user is reponsible for defining the best possible query
filter for the application. Superbase will follow its own internal logic to obtain
optimum performance. However, if you have a complex application involving
many files, you should be prepared to try out various forms of filter line and
see which is the most efficient. Since queries can be stored on disk, this does
not involve repeated redefinition of every element of the query.

One area in which experimentation is often necessary is the order in which
files are referred to in the Query Filter line. Files are processed in the order in
which they are referred to. In a typical one-to-many application it is usually
more efficient to process the file containing the 'ones' first, so you should
place its common reference field on the left-hand side of the first join
expression:

>    Customer_ref.Clients = Customer_Ref.Trans

This forces Superbase to read the first record from the **Clients** file, then find
all the matching records from the **Trans** file, then get the next **Clients** record,
and so on. Provided there are more transactions than clients and you want to
process all the clients anyway, this should be the most efficient filter – the one
that requires the fewest disk accesses. However, in some cases the proportions
of the data may be the other way round: you are restricting the range of clients,
or there are few transactions to process; or both conditions may apply. In this
case it may be more efficient to process the **Trans** file first.

## Optimization with Operators

Not all of the operators available for use in filter command lines allow
Superbase to optimize its own performance. Basically, Superbase needs you to
define either an equality or a range for an indexed field. This technique allows
Superbase to read only the relevant section of the file in question.

The range may be defined as within a file or as either end of a file:

>    > "Bill" AND < "Fred"
>    >= "Bill" AND <= "Fred"
>    < "Fred"
>    >= "Fred"
>    <= "Fred"
>    > "Fred"

Expressions involving the LIKE operator cannot be used in optimization, but
do not actually stop it. However, any occurrence of OR, NOT or parentheses
prevents Superbase from optimizing the remaining expressions in the line.
Note that all Superbase functions use parentheses.

## Using OR in the Query Filter Line

The OR keyword prevents further optimization of the query. It must not be used to relate join expressions, and it should not be placed before any part of a filter command line which Superbase could optimize, such as the range expressions above. If you want to use an OR expression later in the line, enclose it within parentheses:

( City.Clients = "London" OR City.Clients = "New York" )

## Sequencing Precision

A further optimization consideration is the correct sequencing of expressions in the filter command line. The rule states that you should sequence all filter expressions in an order from most precise to least precise. For example:

Trading.Clients = Trading.Trans AND Country.Clients = "USA"
AND Trading_balance.Clients > 1000 AND ( Lastname.Clients
LIKE "smith" OR Lastname.Clients LIKE "jones")

The join comes first, followed by a equality match on 'USA' (case sensitive), which as a single value is precise. Then comes a GREATER THAN expression – less precise, as many values can be greater than 1000. OR expressions, whatever the degree of precision involved, always come last, inside parentheses.

## Defining a Multi-File Query Order Line

The Query Order line for a multi-file query needs no special treatment. It is of course necessary to ensure that the order line includes all fields on which the query output is grouped, as specified by the GROUP keyword in the Query Report line.

More specifically, the Query Order line must correspond to the group structure which you establish in the Query Report line. If you have defined two levels of grouping, the Order line should sort the query output on the basis of the two group fields.

For example, suppose your Report line contained the commands:

GROUP Country.Clients, GROUP Trans_type.Trans

For this report to work correctly, the Order line must be:

Country.Clients, Trans_type.Trans

The procedure for defining a Query Order line is as follows:

1.   Click the Order button in the Query Definition requester.

2.   Click the file selector button next to the file name to select the file from which you wish to choose fields.

3.   Select the field on wish you wish to sort.

4.  Click either the Ascending or Descending button.

5.  Repeat steps 2 to 4 for all the files on which you wish to sort.

6.  Click OK to return to the main Query Definition requester.

# Defining and Using Multi-File Updates

Multi-file updates allow you to update one file on the basis of the contents of another. As such, Update is a powerful tool for making global changes to file data in a single operation.

We will use a specific example to illustrate the procedure for defining a multi-file update. It involves two files called **Employee** and **Payscale**. The first contains details of all the employees in a company, including an employee's name, salary and grade within the company.

Salaries are calculated on the basis of an employee's grade, and the **Payscale** file stores the salary figures for all the different grades. Each record in the file hold just two items of information: a grade and the yearly salary for that grade.

There are two circumstances which require large-scale modifications to the data in the **Employee** file. The company conducts a progress review every six months – and upgrades some employees on the basis of their performance – and, once a year, it also revises the pay scales in line with the rate of inflation.

In a manual system, each of these events would entail a large number of changes to records in both files. With this example of a simple personnel application, you would still need to edit individual records in the both files – to reflect changes in employees' grades and changes to payscales – but salary updates can be carried out automatically using a multi-file update.

The next two sections use this example to describe the exact procedure for defining the multi-file update. See also Chapter 19 Updating.

## Setting the Update Filter

If you are updating one file on the basis of the contents of another file, you must specify a link between the files using the Update Filter.

The first step is to select Update from the Process menu. Superbase displays the Update Filter requester with a list of the fields from the current file. To select another open file, click on the file selector button next to the file name.

In the example application given above, you would now enter the following command line in the Update Filter's Text box:

Grade.Employee = Grade.Payscale

This ensures that the **Employee** file will only be updated with values from the **Payscale** file when the **Grade** field in a record in the **Employee** file is EQUAL

TO the **Grade** field in the Customer file. In another application, you would probably include other conditions in the Filter command line to limit the updating by date or some other factor.

## Setting the Update Definition

Once you click OK in the Update Filter requester, Superbase displays the Update Definition requester. Here you specify the updating action required. In the example, we want the **Salary** field in the **Employee** file to be updated by the **Salary** field in the **Payscale** file; so the command line should be:

Salary.Employee = Salary.Payscale

After specifying the Update Definition command line, click OK to initiate the update processing.

If we continue with the example, we can describe the precise steps Superbase takes when performing a multi-file update. Starting with the first record (according to the current index) in the **Employee** file, Superbase:

1.  Selects the record in the **Payscale** file which has the same grade; that is, the value in the **Grade** field is the same in both files.

2.  Copies the value from the **Salary** field in the **Payscale** file to the corresponding field in the **Employee** file.

3.  Saves the updated **Employee** record to disk.

4.  Selects the next record in the **Employee** file and repeats these steps until all the records in the file have been updated.

## Optimizing an Update

If the first field that occurs in the Update Filter line is an indexed field, Superbase will make use of the index when selecting records. With a large file, this may speed up the operation considerably. The index must be the current index and should be selected (using Index Open or the '–' key) before you select the Update command.

However, you should not attempt to optimize an update in this way if the indexed field also occurs in the Update Definition command line. See Chapter 19 Updating for more details.

# 18 LINKING FILES

Superbase files can be linked in three situations.

- Queries and Updates may include relational links, which are used to join files together for the duration of the processing only. The link information may be stored in the query or update file. See Chapter 17 Multi-File Applications.

- In Forms, files may be linked to provide relational browsing capability. This allows Superbase to retrieve sets of related records from multiple files as you step through one of them. The link information is stored in the Form file. See Volume 2, Form Designer User Guide, and Chapter 10 Retrieving Data Interactively.

- In data entry and editing, the LOOKUP function may be used to cross-check data as it is entered into one file against data in another; a REQUEST command can invoke a LOOKUP requester showing records from another file, from which you can pick the required data. The link information is stored in the File Definition.

The previous chapter discussed the general principles involved in setting up multiple file applications. Here we describe the two Superbase commands – LOOKUP and REQUEST – which are used to establish links between files within a file definition.

You may use LOOKUP in two ways:

- For cross-file validation, where it checks whether the contents of a field in one file match the contents of a field in another file.

- Within a calculation formula, to extract data from another file.

REQUEST 20 can be combined with LOOKUP in a validation formula to generate a LOOKUP requester.

## Cross-file Validation

The syntax for LOOKUP is:

LOOKUP(field1.File1,field2.File2)

**field1** represents the name of the field to which the validation formula is attached, i.e., the current field. **field2** represents the name of the field in another file.

For example, you could use LOOKUP to validate the **Country** field in the **Clients** file by checking it against the **Name** field in the **Country** file. The formula to use would be:

LOOKUP(Country.Clients,Name.Country)

When you enter data in the **Country.Clients** field, LOOKUP will check the records in the **Country** file to see whether the name of the country exists. If it finds a record where **Name.Country** matches the name you have entered it accepts the entry. If the name does not exists, it rejects your entry and displays a validation error message.

There are a number of rules governing the way LOOKUP is entered in a validation formula:

*   The second field name must be the name of an indexed field in another open file. LOOKUP cannot be used for single file validation.

*   LOOKUP will not work with date/time and numeric fields that have null values; i.e., fields that have been left blank. If you wish to use LOOKUP on fields of this type you must use a string function to force them to give an empty string instead of a null value. For example:

    LTRIM$(DATE$(date field))

    will give an empty string when 'date field' has a null value.

*   LOOKUP is case sensitive, so there must be an exact match between the data in both fields; i.e., if the data in the first field begins with a capital letter, the second field must also be capitalized. See the section below.

## Text Field Attributes

If you are validating data entry to a text field by checking it against data in another file (using the LOOKUP function), it is important that the data in both fields should be in the same case. The best of way of ensuring an exact match between data in both fields is to set the same text attributes in their respective file definitions.

The text field attributes are set in the Text Format requester. In most cases, you will specify one of the text field attributes when you first define the file. But you can also do this at a later stage using the Project Modify File command.

The five relevant text field attributes are represented by the first five buttons below the field length counter: Standard, Upper Case, Lower Case, Capitalize Field, Capitalize Word. Click the button for the attribute required.

Standard is the default attribute and leaves text as it is. If you select this option, text is stored and displayed in the case in which it is entered. The other options convert text to a specified format.

*   Upper Case converts all the letters in a text entry to capital letters.
*   Lower Case puts all the letters in lower case.
*   Capitalize Field makes the first letter in the first word a capital letter.
*   Capitalize Words capitalizes the first letter of each word in the field.

## LOOKUP Requesters

LOOKUP requesters are an extension of Validation help messages (see Chapter 7 Data Entry Validation). There are some circumstances in which a single line message is not in fact very helpful. An example would be the following validation formula which checks the data you enter against the data in another file:

> LOOKUP(Stockcode.Invline,Stockcode.Invntory) ELSE "Stock code not found in Inventory file"

When you enter the code for a stock item into the **Invline** file, the formula uses the LOOKUP function to check whether the code exists in the **Invntory** file. The trouble is that if users make a mistake, it is probably because they are not sure what the code is for a particular stock item. Ideally, they should be shown a list of all the stock items in the file together with their codes.

This is just what LOOKUP requesters do. They display a list of all the acceptable data inputs, and allow the user to select an item by clicking on it (without having to type it in).

To display a help requester, you need to use the REQUEST command in your validation formula. For the example above, the formula would be:

> LOOKUP(Stockcode.Invline,Stockcode.Invntory) ELSE REQUEST "Stock code not found","Select another stock code", 20,, Stockcode.Invline,40,Stockcode.Invntory, Description.Invntory

When the user enters the wrong stock code, the following requester will appear:
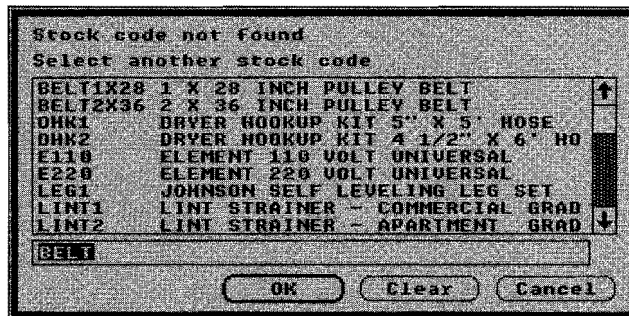


Provided the first field is an indexed field, typing the first letter of the stock code would bring up all the stock codes beginning with that letter. Alternatively, you can use the scroll bars to scroll down the list.

Double clicking on any item in the list automatically places the item into the field specified (in this example, the **Stockcode.Invline** field).

## REQUEST 20

When you create a validation help requester, it is important that you enter the correct parameters for the REQUEST 20 command.

The command itself is an example of the way in which the Superbase Database Management Language can be accessed from a higher level of the Superbase interface.

The first eight parameters are compulsory and must be separated from each other by commas. You may also add two optional parameters at the end of the REQUEST command, and these must also be separated by commas.

1.  For the first two parameters following REQUEST , enter the message that is to appear at the top of the validation help requester. If you wish, you can leave these parameters blank by entering two empty strings; i.e., the command would start with:

    REQUEST "","", ...

2.  The next parameter, **20**, does not make any difference to the way the LOOKUP requester functions, but it must be entered in the formula. Note also that 20 must be followed by two commas.

3.  Now enter the name of the field to which the validation formula is attached. When you select an item using a REQUEST requester, the item will be entered into this field.

4.  The next parameter (**40** in the example in the previous section) specifies the width of the requester in terms of the number of characters.

5.  Enter the name of the field which will appear on the left of the requester box list. This field (**Stockcode.Invntory** in the example) is the one which will be selected for data entry.

6.  As an option, you may also specify up to two more fields, which will be displayed in the requester to the right of the field specified in the previous step. These allow you to provide the user with additional information about the data.

**Note**    The REQUEST command has several other forms. Some of these are also permitted in a validation formula. See Volume 2, DML Reference Guide for full details of the syntax of REQUEST .

# Cross-file Calculation

In cross-file validation, LOOKUP checks the data you enter at the keyboard against the data in another file. In cross-file calculation, it automatically extracts data from another file.

To understand what is involved in this process, you need to be clear about the the idea of the current record. If a cross-file validation is successful – if LOOKUP finds a matching field in another file – the record containing the field becomes the current record, even though the file may not be the current file and is not displayed on screen.

Each open file has a current record. The current record for the current file is simply the one that is displayed on the screen. Although the other files are behind the scenes, you can access their fields in the same way as with the current file. When you refer to a field in a file other than the current file, Superbase looks for data in the current record for that file.

These features provide a way of inserting the data from another file in a new record, using LOOKUP in a ternary calculation formula (see **Using the Ternary Operator**, Chapter 8 Derived Values). We can illustrate this with an example involving two files, **Orders** and **Invntory**.

Each record in the **Invntory** file contains details of a particular product, such as its stock code, its price and a description of the product. **Orders** contains details of a customer's order; these include a stock code for each product item ordered and the quantity of items ordered. Note that the stock code field is common to both files and provides the relational link between them.

When a new record is created in the **Orders** file (or when an existing record is edited), the stock code and quantity will have to be typed in. But by using LOOKUP in combination with the ternary operator, we can ensure that the price (the unit cost) of a product and its description are automatically read into the new record from the **Invntory** file.

The calculation formula to attach to the **Unit_Cost.Orders** field would be:

```
LOOKUP(Stockcode.Orders, Stockcode.Invntory) ?
Unit_Cost.Invntory: 0
```

The effect of this formula is to force a LOOKUP on the **Invntory** file whenever new data is entered (or existing data is edited) in the **Unit_Cost.Orders** field. First it checks to see whether the stock code in the current record in the **Orders** file exists in the **Invntory** file. If it does exist, it makes the matching record in the **Invntory** file current. The condition expressed by the ternary operator has been satisfied, so the first value after the question mark – i.e. the contents of **Unit_Cost.Invntory** – is assigned to **Unit_Cost.Orders.**

If the condition is not satisfied, the second value – in this case, a value of zero – is assigned. With a string field, the second value must be a text string, so the calculation formula for the **Description.Orders** field would be:

```
LOOKUP(Stockcode.Orders, Stockcode.Invntory) ?
Description.Invntory: ""
```

Here, the empty string is assigned to **Description.Orders** if the stock code doesn't exist in the **Invntory** file; in other words, the field remains blank.

It is up to you to decide what action is to be taken if the LOOKUP is unsuccessful. With string fields, you may want to assign a text string indicating that an error has occurred, as in:

```
LOOKUP(Stockcode.Orders, Stockcode.Invntory) ?
Description.Invntory: "Error"
```

Now if there is no match between the stock codes in the **Orders** and **Invntory** files, 'Error' will be assigned to the **Description.Orders** field. Normally, however, you would also use LOOKUP in a validation formula attached to the **Stockcode.Orders** field, so the situation would never arise: the stock code in the **Orders** file will always be matched with a stock code in the **Invntory** file and, as a result, the condition in the calculation formula will always be satisfied.

## Cross-File Calculations and Forms

If a form refers to more than one file, you have to define a link structure in the Form Designer that will allow Superbase to retrieve related records correctly. Since the separate files are linked at the form level, some cross-file calculations may become redundant. For example, there is no need to use a LOOKUP in a ternary calculation formula to force specific records to be read; the link structure does this automatically.

In some ways, using forms allows you to simplify your file definitions. You do not need to duplicate information from one file into another. In the example in the previous section, a formula is used to copy the **Unit_cost.Invntory** field into the **Unit_Cost.Orders** field. But in a form, although you must include a LOOKUP to access the **Invntory** file, fields from that file may be displayed and referred to independently.

# 19 UPDATING

The Update command on the Process menu allows you to make changes to individual fields in your files. Update provides a multi-file facility, like Query, and you can use records in one file to determine what happens to fields in another. For information on relational updating, see Chapter 17 Multi-File Applications.

## The Update Menu Commands

There are three Update menu commands: Edit, Open and Save As. You use Edit to create a new update, or to edit an update which has been loaded from disk. Open is used to load an update from disk, Save As stores an update on disk. When you save an update file, Superbase also saves the names of the database files associated with it. These files will then be opened automatically when the update file is next loaded. Update files are stored on disk with the extension SBU.

## Starting a New Update

Update can use only the open files, so if you want to alter or refer to a field using this command you must first make sure the file that contains the field is open. Use the Project Open File command to do this.

Now select Update from the Process menu. Superbase displays the Update Filter requester.
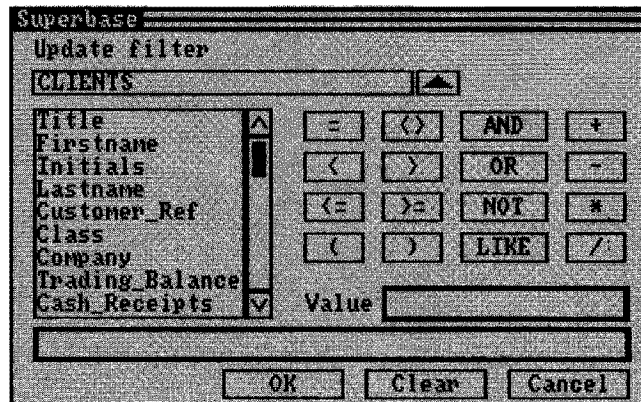
## Selecting File and Fields

The Update Filter requester is like the standard browsing controls filter, except that there is an arrow button to the right of the current file above the List box.

When you are performing a relational update, this arrow enables you to display the fields from all the open files in turn. Each time you click on it, the fields from the next open file are displayed in the List box below. If you go on clicking, the list of files cycles back to its beginning.

## Setting the Update Filter

If you want to update all the records in the file, click OK without entering anything in the Text box.

Otherwise, you can use the Update Filter to restrict the update to a specific range of fields. Set the filter in the normal way, by clicking on field names and operators, and entering values in the Value box. Superbase copies the results of your selections into the Text box below.

You can edit the Text box, but since this increases the risk of errors, you should only do this when you have some experience of the system.

The Filter command line can contain any combination of field names from the open files, and can have a maximum length of 512 characters. When you have created the Filter you want, click OK. Superbase then displays the Update Definition requester.
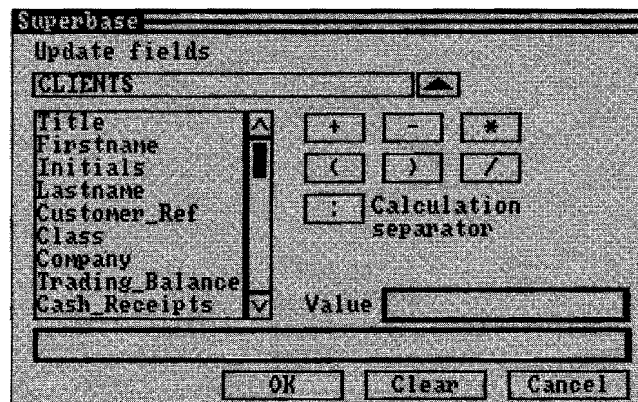
## The Update Definition Requester

In this requester, you specify the precise updating actions you want Superbase to perform on certain fields. Typical updating requirements might be

- Setting a numeric field to zero.

- Adding transaction amounts to balances.

- Setting a status indicator field to a new value.

Each updating action is defined the same way. An example:

Code = "Y"

This sets the **Code** field to the value 'Y', irrespective of its previous value. Another example:

Balance = Balance + Amount

Here the **Balance** field is mentioned twice. This has the effect of adding the **Amount** field to the **Balance**, instead of replacing the old value of **Balance** with the new value of **Amount**. The third example increases the **Amount** field by ten percent:

Amount = Amount * 1.1

To define an update command in the Update Definition requester:

1.  Click the name of the field you wish to update. Superbase copies it into the Text box, and automatically adds an EQUALS sign operator.

2.  Enter the formula or expression which will calculate the value that is to be assigned to the field on the left of the EQUALS sign. You may create an expression using any combination of fields, constant values, and operators up to a maximum length of 512 characters.

    □   If you wish to include a field from another file, clicking the arrow button to the right of the file name cycles through all the open files and displays their fields.

    □   Click in the Value box to enter a numeric or string value, remembering to press ENTER as you finish.

    □   To enter an operator or the colon separator (see below) click its button.

3.  Click OK to complete the command line and initiate the Update processing.

## Using the Colon Separator

Each time you enter an update expression, you must set it off from the previous one by clicking on the Colon button. This inserts a colon in the command line. If the examples above were all in one line, they would be separated like this:

Code = "Y" : Balance = Balance + Amount : Amount = Amount * 1.1

## Forcing Calculations to Evaluate

The Update command provides a simple way of forcing Superbase to evaluate a calculation which is assigned to a field in the file definition. Typically, you would want to do this for all the records in a file after modifying the file definition; either because you have changed an existing calculation formula or because you have added a new field with a calculation attached to it.

The procedure is as follows:

1.  Make sure the file you have modified is the current file.

2.  Select Update from the Process menu.

3.  When the Update Filter requester appears, leave the Filter command line blank and click OK or press ENTER .

4.  In the Update Definition requester, enter the name of the field you wish to evaluate, followed by an EQUALS sign. Then enter the same field name again. For example, to force evaluation on the **Balance.Clients field**, you would enter:

    Balance.Clients = Balance.Clients

5.  Click OK. Superbase now evaluates the calculation for all the records in the file.

# 20 REMOVING RECORDS

## Removing Records with Cut

When you need to delete an individual record from a file, use the Cut
command on the Edit menu. This command always works on the current file
only – the one whose name appears in the window title bar – so if you are
working with a multi-file form and you choose Cut, only the data from one of
the files will be removed. If you need to remove data from more than one file
when a form is open, switch to each file in turn and use Cut. Making a
transaction file – one which is the 'many' of a one-to-many relationship – the
current file ensures that only one record at a time can be deleted from it. This
is a precaution against accidental destruction of data.

See Chapter 6 Entering Data for further information.

## Removing All Records

With the Remove command on the Process menu, you have the option of
removing all the records in the current file or set of records specified with a
filter.

*   If a multi-file form is open, Process Remove will only remove records
    from the current file, as indicated in the window title bar.

1.  Open the file from which you wish to remove records. Make sure that it is
    the current file.

2.  Select Remove from the Process menu. Superbase displays the Remove
    Records Filter requester.
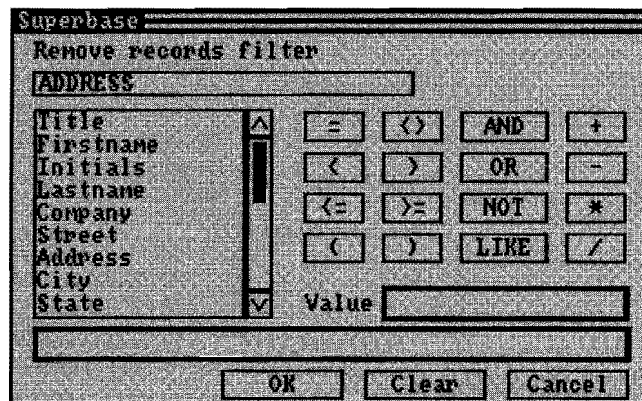
3. Click OK in the Filter requester, leaving the Text box empty.

4. Superbase displays a confirmation requester. Click OK to proceed, Cancel to abandon the action.

5. If you click OK, Superbase will remove all the records in the file by initializing the file.

**Caution** Data removed is not recoverable.

# Removing Selected Records

1. Open the file from which you wish to remove records. Make sure it is the current file.

2. Select Remove from the Process menu. Superbase displays the Remove Records Filter requester.

3. Enter a filter which specifies the conditions under which a record is to be removed. Build up the Filter command line in the normal way.

4. Click OK in the Filter requester.

5. Click OK in the confirmation requester.

## Example

In this example we'll use the **Clients** file, so make it the current file if you haven't already done so. Then select Process Remove.

Your objective is to remove from the file all the addresses where the **Country** field has the value 'England'. Click **Country**. Superbase copies the field name into the Text box. Now click the LIKE operator. Then click in the Value box, and type 'England' (omit quotation marks). Press ENTER . Your Filter command line should read:

    Country.Clients LIKE "England"

If it doesn't then click the Clear button to clear the Filter, and start again. (If you click Cancel it will have the same effect, but Superbase will assume you've finished with the Filter and remove the requester from the screen.) When you've got the Filter command line right, click OK. Superbase now checks the Filter command line to see whether there are any errors in it. If you've entered the example correctly there shouldn't be any.

Next comes a confirmation requester, as Superbase does not allow records to be removed with a single action. Click OK to go ahead, Cancel to abandon the action. Provided you click OK, Superbase removes all the records that match the Filter. The Filter command line is then cleared out, and Superbase returns to the Work Area.

# 21 IMPORT

## Overview

There are only two ways in which data can be entered into Superbase's own format. The first way is through the keyboard, using a Record or Page View, a Form created using the Form Designer, or a DML program. The second way is automatic, through use of the Import command. Import is selected from the Process menu and is also available as a command in DML.

As computers become more and more a part of business and professional life, so the amount and variety of data stored on computers keeps on growing. Much of this data is readily accessible to you via the Import command. Data files can be downloaded from telecommunications-based information services, transferred directly via modem from another computer, or via floppy disk or magtape. Most information providers are now offering services based on at least one of these methods.

Data can also be captured automatically from documents using an optical reader or scanner combined with character recognition software. This technique, called OCR, typically converts bit-mapped input into ASCII characters.

Using the Import facilities offered by Superbase, data obtained in any one of these ways can normally be converted into the Superbase format. You can then use Superbase to carry out searches and selections on it, reformat all or part of it, integrate it with other data and use it in your own database application.

As a further step, many of these operations can be completely automated using the communications and other commands available within DML.

### When to Import dBase Files

In addition to its own format, Superbase can view and report directly on data maintained externally in dBase II or dBase III format. Superbase cannot update data in dBase format, only data stored in its own format. This safeguard is necessary in order to prevent Superbase users from inadvertently violating data integrity or contention checks programmed into dBase applications.

Superbase does however provide powerful facilities both for importing and for exporting files in dBase II or dBase III format. If your application calls for the updating of data stored externally in dBase format, you should first import the data, then carry out the required updates, then re-export the data back into its original dBase format using the Export command (see Chapter 22 Export).

Batch-oriented tasks of this nature can be completely automated using DML.

## Uses of the Import command

In addition to dBase format files, the Import command also includes options for importing many other popular file formats. The Import and Export commands both support the same range of file formats, facilitating the ready transfer of data in either direction between Superbase and other applications.

The Import command is used in three distinct ways:

- As a file conversion utility. In one step it converts a file or a selected part of a file created by another application into a new Superbase file, complete with its associated file definition and index. Import can convert files in many of the popular database and spreadsheet formats. It can also convert files in ASCII delimited format.

- As a merge utility to merge data stored in fixed length or delimited ASCII format into a pre-existing Superbase file. Almost all applications are capable of producing ASCII files as output. You can accordingly use this method for transferring data between applications where it cannot be directly converted by the first method.

- As a merge utility to merge two Superbase files.

## Import File Types

Superbase can convert files of the following types into new Superbase files:

| Application | Filename Extension |
|---|---|
| ASCII delimited | any |
| dBase II, III, III+ | DBF |
| Enable | DBF |
| Lotus 1-2-3 | WKS, WK1 |
| Microsoft Excel | XLS |
| Grafox Logistix | LGX |
| VisiCalc and others | DIF |
| Superplan | SPP |

**Note**    The character set used by DIF files is assumed to be the IBM character set, unless the ANSI conversion option has been selected or the header of the DIF file has the title 'Excel'.

Superbase can merge files of the following types using an existing Superbase file definition:

| Application | Filename Extension |
|---|---|
| ASCII fixed length | any |
| ASCII delimited | any |
| Superbase | SBF |

# ASCII Files

If you intend to use the Import command on an ASCII file, you should first ensure that it is indeed an ASCII file, and not one containing embedded nulls or control characters. The existence of such characters will cause Import to malfunction. It is not sufficient to use the Type command or the Text Editor to read the file. These utilities will ignore control characters and nulls in displaying the file contents. If in doubt, you can view the file using the CLI 'Type' commands.

Non-ASCII files can often be merged into Superbase files under DML control using the OPEN FOR INPUT and INPUT commands. If you attempt this, bear in mind that Superbase treats a null (ASCII Code 00) as a string terminator.

Once you are sure that the file you are dealing with is an ASCII file, you need then to find out whether it is a delimited or fixed length file, and if delimited the values of the field and record delimiters.

## ASCII Delimited Files

In ASCII delimited format, the data is stored as ASCII characters with further special characters to mark the end of each data item. These are called delimiters or separators. The data should be organized sequentially into records, with a delimiter separating each field in the record including blank fields, and a different delimiter separating each record.

Part of a typical ASCII delimited file might look like this when viewed by the Type utility or the Text Editor:

Mr.,John,Smith,Carpenter,415 345 6789
Mrs.,Susan,Jones,Cook,408 546 3456
,Julian,Doe,,415 465 6781

This section contains three records, each of which holds five fields: a title, first name, surname, profession and a telephone number. Notice how the last of the three records has a blank field for the title and further blank field for the profession. These fields, though blank, are still terminated by separators.

**Caution**  ASCII delimited files must contain the right number of separators for each record, or the data will be imported into the wrong fields.

## ASCII Delimited File Separators

In the above example the field separator is the comma (ASCII code 44), and the record separator the carriage return character followed by the line feed character (ASCII codes 13 and 10). These are actually the default separators assumed by Superbase.

If the separators in your ASCII file differ from these, select System Options on the Set menu to alter the default separator values accordingly. Up to two characters may be used for each kind of separator.

Characters are specified by typing in the ASCII code equivalent to the character you want to use. A list of ASCII codes is provided in Appendix D. To change the default values, move the pointer to the relevant box and click in it. This will produce the insertion point and you can then type in the new ASCII code. To enter a two character separator, type in a comma between the ASCII codes for the two characters.

**Caution**   The same characters should not be used for both field and record separators.

## Quotation Marks

ASCII files may well contain as data one or more of the characters normally used as separators, for example names and addresses typically include commas. For this reason Superbase provides an option during Export to enclose each field in double quotation marks:

```
"Mr.","John","Smith","345, Express Highway","415 345 6789"
"Mrs.","Susan","Jones","45, Juniper Drive","408 546 3456"
"","Julian","Doe","","4154656781"
```

This option can be set using the System Options command on the Set menu. In the System Options requester, click the check box next to the Use Quotes option.

Superbase will automatically discard double quotation marks around fields when it finds them during an Import.

## ASCII Fixed Length Files

In ASCII fixed length files, each field or record occupies a fixed amount of space on disk. Each record contains the same number of characters or bytes, and a given field contains the same number of characters for each record. The data is stored as a continuous string of ASCII characters including spaces.

Some ASCII data files consist of fixed length records separated by carriage returns (mainframe software often produces this kind of output).

# Converting Files

## Converting ASCII Delimited Files

ASCII fixed length files can only be imported by merging with an existing file definition. This operation is covered below. Superbase can however convert ASCII delimited files into a new Superbase file, using the field delimiters as a basis for the file definition it creates. Data types are assumed as Text, and Field names given arbitrary values by the system of the form A,B..,AA,BB..etc. Following completion of the Import process, Superbase invokes the File Modify requester, displaying the arbitrary field names and other file definition parameters for amendment.

## Converting Database Files

Superbase converts files in DBF format into new Superbase files. Following selection of this file type, Superbase will ask you to select a file from a requester displaying all the filenames in the current directory with the DBF extension. It will then automatically create the SBF file and SBD file definition, taking the SBD file definition field names and types from the DBF file header. Finally Superbase displays the New Index requester and requests selection of at least one field for indexing purposes. For each field selected it creates an index to the new file.

Superbase is designed to access records in index sequence. Individual records in dBase filing systems are often identified by record number. If you wish to continue identifying the records in your imported file in this way, you should ensure before importing that the dBase file includes a field containing this record number, since Superbase can only create an index on an existing field.

## Converting Spreadsheet Files

Superbase can convert data in various spreadsheet formats into new Superbase files. Supported formats include WKS, WK1, XLS, LGX, SPP and DIF.

When importing spreadsheet data, each row is taken as a record, and each cell as a field in the record. Columns that are completely empty in the spreadsheet are ignored. The spreadsheet data should accordingly be structured as a series of repeated rows.

Superbase provides the option to select a rectangular block of cells to import from the spreadsheet by specifying the range from row/column to row/column. It is also possible to import the Superbase field names from the spreadsheet file by specifing on which row number in the spreadsheet these labels appear. If a labels row number is not specified, each field name is made up from the spreadsheet's column letter followed by an underscore character, for example 'A_'.

# Merging Files

## General Considerations

Superbase can merge data from an ASCII delimited or fixed length file or from a Superbase file. It always merges to an existing Superbase file. The 'merge to' or destination file may or may not contain pre-existing data, but must have a valid SBD file definition. The Import process will always attempt to create new records in the destination file rather than update existing ones.

Import assumes that the destination file is the current open file. If the fields in the source file are not in the same sequence as the fields in the destination file, or if there are more fields in the destination file than there are in the source file, you should make a Field Selection on the current open file, selecting the necessary fields in the sequence that corresponds to the source file.

If you are Importing into a Field Selection, then Superbase will create new records with the fields not selected left blank. Such fields should not be indexed, allowing only unique keys, as otherwise the Import will fail when Superbase attempts to create the second blank index key.

Superbase will apply to the incoming data whatever data typing, conversion, validation and calculation rules are specified in the destination file definition. If any one of these rules does not apply to the incoming data it should be removed or relaxed using Project Modify File before the Import takes place. Otherwise Superbase will not be able to successfully complete the Import.

For example:

- Required fields must contain data. If an incoming record has no value for a required field, Superbase will signal an error.
- Validations in the current file definition will be carried out before the record is stored, whether the field value has been imported or not. If a validation does not yield the correct result, Superbase will signal an error.
- Calculated, Constant and Virtual fields are derived from the current file definition rather than input. Values in corresponding fields in incoming records will thus be disregarded.
- If a text field definition in the destination file specifies upper case only for example, the incoming data will be converted accordingly.

The data types of fields in the source and destination files should always correspond. Superbase will convert incoming ASCII text strings containing numeric or date information into Superbase numeric or date fields, on condition that the same values would have been accepted during manual data entry.

### Merging ASCII Files

If you are importing an ASCII file into an existing Superbase file, you can set a filter to determine which records are to be imported. The filter is specified using the destination field list. See Chapter 10 Retrieving Data Interactively for a detailed explanation on how filters work.

The delimiters in an ASCII delimited file must correspond to the values set in the System Options requester. See above for more information about ASCII delimited files, separators and quotation marks.

With ASCII fixed length source files, the field lengths during import are assumed to be the lengths set in the destination file definition. If in doubt, inspect the source file using Type or the Text Editor in order to determine the field lengths, then use File Modify on the destination file to set its field lengths accordingly. Before importing a fixed length ASCII file that includes record separators, turn on the Fixed Length Separators check box on the System Options requester, and specify the record separator in the normal way. See above for more information about types of ASCII file and ASCII fixed length files.

# Using Process Import

Before operating the Import command you should know the format of the file you are importing from, and whether you are going to be creating a new file or merging imported data into an existing file. You should then read the relevant sections above for detailed instructions about this type of import operation.

If merging, make the destination file the current file and open a Field Selection list, if appropriate.

1.  Select Import from the Process menu. Superbase displays the following requester:

2. The upper part of the requester shows the three merge options. If you are merging, select one of these. Otherwise select the type of file you wish to import to create a new file. Click on the appropriate button.

3. If you have specified one of the spreadsheet types, you now have the option of specifying the range of cells to be imported and the number of the spreadsheet row containing the labels to be used for field names.

4. Once you are satisfied with your selections at steps 2 and 3 click OK. Otherwise click Cancel to return to the main menu.

5. If you are merging an imported ASCII file you now have the option of specifying a filter on the file. Formulate the filter and click OK or click Cancel to return to the Import requester.

   □ The source file must be opened first to ensure that field references are understood.

6. You will now see a file selection requester. Go to the directory containing the source file. If you have specified ASCII format, the requester lists all files in the current directory. Otherwise it shows only those files with the corresponding extension, for example DBF. Select the file to be imported and click OK, or click Cancel to return to the Import requester.

7. Once the processing of the source file is complete, if the Import process creates a new file, you will now see a New Index requester. Select the field or fields on which the new file is to be indexed.

8. If you have created a new file from an ASCII delimited source file, you will now see the File Modify requester for the new file. Here you can replace the arbitrarily assigned field names with meaningful field names of your own.

# 22 EXPORT

The Process Export command allows you to convert a Superbase file to a
different format so that it can be loaded into another software package.
Generally, you will export files for use in a spreadsheet or another database
program, but you may also want to load Superbase data into a wordprocessing
program or a desktop publishing package.

The only restriction is that the exported file must be in a format that is
recognized by the other program. This should not be a problem since Export
offers the same range of non-Superbase formats or file types as Import,
including two ASCII file types. Even if the other program does not provide any
file conversion facilities, it will almost certainly be able to accept ASCII files.

## Preparing to Export a File

Before you select Export, you can specify which fields are to be exported, using
the Field Selection command on the Set menu. In creating new records
Superbase will only take the data from the selected fields list.

Records are exported in the order of the current index, so you should select
the index you require before selecting the Export command; use the Index
Open command or the '−' key on the numeric keypad.

You also have the choice of exporting records in ascending or descending
order. By default, records are exported in ascending order. Click in the check
box labelled Descending in the Export File Type requester (see below), if you
want to export records in descending order.

If you intend to convert the Superbase file into an ASCII delimited file, two
further preliminary commands are available:

- Field and Record separators. Depending on the application, you may
  want to change the default values for the field and record separators. To
  do this, select System Options on the Set menu. See the section on ASCII
  delimited files in Chapter 21 Import for more details.

- Quotation marks. The System Options command in the Set menu allows
  you to specify that every field exported is enclosed in quotation marks.
  This is necessary if the data contains the character used as a field
  separator; for example, addresses, dates or numbers may contain commas.
  See the section on ASCII delimited files in Chapter 21 Import for more
  details.

# Using Process Export

1.  Make sure that the file you wish to export is the current file. Then select Export from the Process Menu. Superbase displays the Export Filter requester. Use this to set a filter which restricts the range of records that will exported. If you wish to export all the records in the file, click OK without entering anything in the Text box.

2.  Superbase now displays the Export File Type requester.
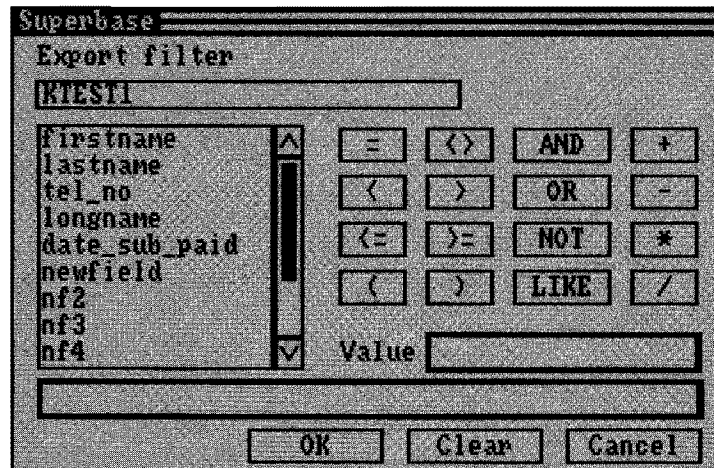


Use this to specify how the file is to be exported. Click on the button corresponding to the file type required.

□  Details of the different file types are given in Chapter 21 Import. Here, we only need to mention that the fields in ASCII fixed length files are exported in the format set in the current file definition. If the data in a field is shorter than the length set in the file definition, Superbase pads it out with spaces.

□  Click in the Descending check box to export records in descending order; that is, in reverse order to the current index.

3.  Click the button corresponding to the file type required. If you select a spreadsheet file type, you also have the option of specifying that labels are included – Superbase will then export the field names as a set of labels along the first row of the spreadsheet.

4.  When you click OK, Superbase displays the Export File Name requester. Enter the name the file is to have when it has been converted.

☐ With ASCII files, you should supply an extension name; we suggest that you adopt a convention for naming exported ASCII files, such as ending each with the extension EXP, or CSV (Comma Separated Value) if the fields are separated by commas.

☐ If you have specified one of the other file types, you should not include the extension name: Superbase will add the appropriate extension for you.

Superbase then processes the database file, checking each record to see whether it matches the conditions set up in the Filter. If it does, Superbase creates a copy of the data in the currently open fields and stores it on disk.

# 23 EDITING TEXT

The Text Editor is suitable for entering and editing text of almost any kind, from short memos and letters to lengthy written reports. Both Superbase's SBT text files and ASCII/ANSI text files are supported. The Text Editor is not, however, intended as a replacement for a word processing program, and it does not provide all the features you would expect to see in a word processor. But you may find it more convenient to use the Text Editor for many of your word processing requirements.

More specifically, it is used for two purposes relating to Superbase itself:

• Creating form letters for Mail Merge applications (see Chapter 25).

• Storing text items which are longer than the maximum length of a text field as external text files linked to database records.

**Note**  Throughout this chapter, we use the term 'document' to refer to the text in memory, and the term 'text file' to refer to a file on disk.

To invoke the Text Editor, select the Text Editor command from the Utilities menu (or press AMIGA+H ). Superbase opens a window in the right-hand part of the screen.

## The Text Editor Window

This is the text window where you enter text and edit it. Four menus are now available: Project, Edit, Search, and Options.

The database window showing the current record remains open but is now inactive. If you move the mouse pointer anywhere in the database window and click in it, the window will become active again. Superbase lets you switch between the Text Editor and the database at any time simply by clicking in their respective windows.
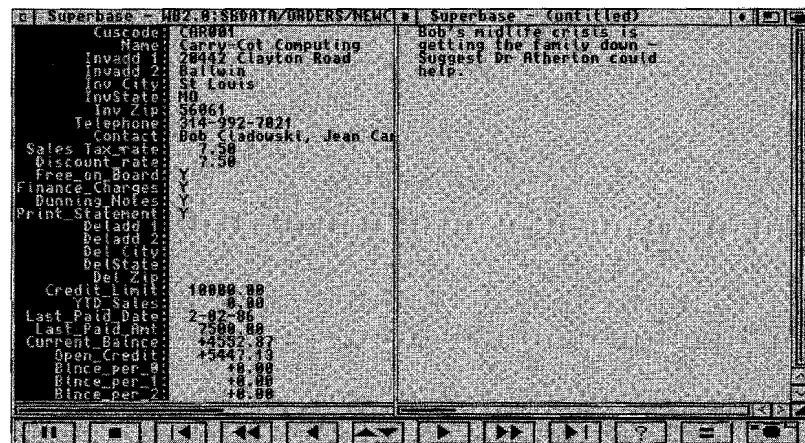
If you are editing a large document, the text window will only show part of it. The scroll bars or direction keys enable you to move the window in any direction over different parts of the document (see Scrolling the Text Window). The maximum width for text is 240 characters.

## Selecting Full Screen

If you prefer to edit text using the full width of the screen, click the Maximize box (the third small box from the right at the top of the text window).

## Closing the Text Window

To close the text window, either click in the box at the top left-hand corner of the text window or select the Exit command on the Project menu. This does not remove text from the computer's memory. If you open the text window later, any text you have entered will still be there.

## Managing Text Files from the Project Menu

Many of the Project menu commands are the same as the Project menu commands in the database window. The difference is that they operate on text files rather than database files.

Most of the time, you will be working with text files you have created in the Text Editor. These are stored on disk as files with a SBT extension. Requesters which present lists of filenames show only SBT unless you alter the file name pattern (see Working with Non-Superbase files, below).

Here's a list of the Project menu commands, together with a brief explanation of their functions:

**New**      Clears any existing text from memory.

**Open**     Loads a text file into the Text Editor from disk. If a document is already present it will be replaced by the new file.

**Close**    Clears a document from memory and closes the text window.

**Save**     Saves a document as a text file on disk.

**Save as**  Saves a named document as a text file on disk giving you the opportunity to rename it.

**Remove**   Removes a text file from disk.

**Print**    Prints the document currently in memory. See 24 Printing Text.

**Exit**     Closes the Text Editor window and returns the user to the Database window. Does not clear the current document.

### Working with Non-Superbase Files

You can choose to work with non-Superbase text files rather than Superbase SBT files. You can tell Superbase to open or save any file as ASCII (or ANSI if you have selected the ANSI File Conversion option in the System Options requester). To open a non-Superbase text file, follow this procedure:

1. Select Open File.

2. In the Text box, delete the SBT extension in the pattern matching string.

3. Type in the extension for the file you wish to open. Alternatively, to view all the files in the current directory, type in the asterisk character.

4. Select the file you wish to open, and click OK.

   ☐ If the file is not readable as ASCII, Superbase informs you.

When the Text Editor has a non-Superbase document in memory, the Reformat, Ruler, Margins, Underline, and Italic menu options are disabled.

If you wish to save an SBT file as an ASCII or ANSI file:

1. Select Save as from the Project menu.

2. Delete the SBT extension and type in a different extension (e.g. TXT).

3. Click OK.

When you convert an SBT file to ASCII or ANSI in this way, Superbase removes all the formatting information – word wrapping, margin settings and text styles – from the file.

# Creating a New Document

When you enter the Text Editor for the first time in a session, you will be presented with a blank page. You can start typing immediately.

If you already have a document in the editor, you should follow these steps:

1. Select the File New command.

   ☐ If the current document has been modified since it was last saved, you will be asked to confirm that you want to go ahead.

2. When the screen is cleared, start typing.

# Entering Text

Text can only be typed in when the Text Editor insertion point is on the screen. The insertion point is a vertical line which shows where the next character you type will appear. If you have just started a new document, the

insertion point will be at the top left-hand corner of the text window, at column 5, which is the default left ruler margin. If the insertion point is not on screen, you can click in the text window to bring it back. You cannot obtain an insertion point below the last line of text.

### Word Wrap

The Text Editor allows you to specify where lines of text start and finish by setting the left and right margins. In a later section, we explain how you can alter the margins for the whole document or each individual paragraph.

By default, the left margin is at column 5 and the right margin is at column 75. If you type a word that extends beyond the right margin, the Text Editor automatically moves the word to start of the next line. This is known as 'word wrap.' The insertion point is also moved to the next line so you can carry on typing without interruption.

### Paragraphs

On a typewriter, you usually need to press the ENTER key when you reach the end of a line. In the Text Editor, you use the ENTER key to end a paragraph. Pressing ENTER takes the insertion point to the start of a new line and a new paragraph.

The Text Editor uses a small black rectangle as a paragraph marker to indicate the line where a paragraph ends. Paragraph markers are placed on the right-hand edge of the text window.

# Editing Text

The Text Editor provides a large number of facilities for editing text, ranging from insert or typeover options to Cut, Copy and Paste.

### Moving the Insertion Point

With the Text Editor, you can move the insertion point around the screen and edit text at any point on the screen.

You can use the mouse to position the insertion point within the text window. Move the pointer to the point where you want to the insertion point to be, and click once. You cannot move the insertion point beyond the last line of text.

There are two ways of bringing text into view if it extends beyond the text window. Moving the insertion point in one direction to the edge of the text window scrolls the window in the same direction. Alternatively, you can scroll the window using the pointer and the scroll bars.

You may prefer not to use the mouse much while you are typing. There is a full set of keyboard equivalents for moving the Insertion Point.

| When you want to: | Press: |
|---|---|
| Move to the left one character | LEFT |
| Move to the right one character | RIGHT |
| Move up one line | UP |
| Move down one line | DOWN |
| Move right to the next multiple of 8 characters | TAB |
| Move left to the previous multiple of 8 characters | SHIFT+TAB |
| Move up one window height | SHIFT+UP |
| Move down one window height | SHIFT+DOWN |
| Move to the end of the line | SHIFT+RIGHT |
| Move to the beginning of the line | SHIFT+LEFT |
| Move to the beginning of the text | CTRL+B |
| Move to the end of the text | CTRL+G |

## Basic Editing Controls

Most simple editing tasks can be performed with these controls only. Some of these tasks are explained in more detail below.

| | |
|---|---|
| Split line | CTRL+S |
| Join lines | CTRL+A |
| Switch between Insert and Typeover | CTRL+V |
| Undo latest edit | CTRL+U |

## Reformatting

The Text Editor automatically reformats the current paragraph whenever you move off the current line. It obeys the margin settings for the paragraph (see below), joins up lines, and wraps words.

You can reformat the text manually by selecting the Edit Reformat command or by pressing CTRL+F .

## Deleting Small Areas of Text

The controls for deleting text operate on the text at the current position of the insertion point. Before using these controls, you should check that the insertion point is in fact located at the point where the text is to be deleted.

| When you want to: | Press: |
|---|---|
| Delete the character before the insertion point | BACKSPACE |
| Delete the character after the insertion point | DEL |
| Delete to end of word | CTRL+W |
| Delete to end of line | CTRL+E |
| Delete current line | CTRL+D |
| Clear current line | CTRL+X |

To delete larger blocks of text, use the Cut facility as explained on the next page.

# Cut and Paste

Superbase's cut and paste facilities let you cut, copy and move blocks of text within a document. Text may also be copied or moved to other documents, or between the Text Editor and database fields.

The first step in a cut and paste operation is to mark the block you wish to work on so that it is highlighted on screen.

## Mark Block

There are four ways of marking a block:

- Double-clicking. Position the mouse pointer over the first letter in the block and double-click the mouse button. Then double-click over the last letter in the block.

- Click-and-drag. Click on the first letter and keep the mouse button pressed down. Then move the pointer to the end of the block and release the mouse button. The area of the block is limited by the borders of the window. If you need to mark a block larger than the window, use one of the other methods.

- Control key. Position the insertion point in front of the first letter and press AMIGA+M . Repeat this step for the last letter in the block.

- Edit Mark Block. Position the insertion point in front of the first letter and select the Edit Mark Block command. Repeat this step for the last letter in the block.

Once a character is marked as a block, repeating the Mark Block operation can have the following results:

- Before the first mark, it extends the start of the block.

- After the first mark, it makes a new end to the block.

## Clear Block

This command removes the block markers and returns the highlighted text to its normal state.

## Cut

This command removes the marked block and reformats the text. The cut text is stored on the Clipboard.

- **Move.** To move a block of text to another part of the document, first highlight the block and select Cut. Then move the insertion point to a new position in the document and select Paste.

## Copy

Copy stores a copy of the marked block on the Clipboard. To copy text, first select the block and select Copy. Then move the insertion point to a new position in the document and select Paste.

## Paste

This command inserts the text from the Clipboard at the insertion point position and reformats the text. Paste does not remove the text from the Clipboard but instead inserts a copy of it your document. This means you can use Paste to make multiple copies of the last item that was cut or copied to the Clipboard.

## Cut, Copy and Paste with Other Files

These commands are also available for use with database files. This means you can freely transfer data between the Text window and the Database window. See Chapter 6 Entering Data.

In addition, you can transfer data between any other Workbench application which provides Cut, Copy and Paste facilities that make use of the Workbench Clipboard (provided the text format is compatible).

When you select Cut or Copy, Superbase places the marked text in the Workbench temporary storage area known as the Clipboard. Paste then inserts the Clipboard text into the current document at the current insertion point. These operations are described in more detail below.

## Cut, Copy, Paste Key Equivalents

Mark Block, Clear Block, Cut, Copy and Paste are available as commands on the Edit menu. The key controls for them are as follows:

| | |
|---|---|
| Mark Block | AMIGA+M |
| Clear Block | AMIGA+K |
| Cut | AMIGA+X |
| Copy | AMIGA+C |
| Paste | AMIGA+V |

# Inserting Text

The way text is inserted in a document depends on which one has been selected of two typing modes: Insert and Typeover. To switch between them,

press CTRL+V . Provided that NUM LOCK is switched off, pressing INS also switches between Insert and Typeover modes. Insert is the default mode.

Each mode has its own type of insertion point. Insert uses a thin vertical line, Typeover uses a thicker vertical line.

Both modes let you add text to the end of a document, creating new lines as you do so. The difference comes when you enter text within a document.

## Insert

With Insert mode, you can insert text in a document without overwriting the existing text. Any text in front of the insertion point is pushed to the right to make room for the new text. At the same time the Text Editor reformats the rest of the paragraph so that it does not extend beyond the right margin.

Pressing either ENTER or CTRL+S in Insert mode splits a line at the insertion point. The text following that point becomes a new paragraph.

To join two paragraphs together, either position the insertion point at the start of the second paragraph and press the Backspace key, or position the insertion point anywhere on the last line of the first paragraph and press CTRL+A .

## Typeover

In Typeover mode, the characters you type in replace the existing characters at the insertion point. Use this mode when you want to rewrite a line or a paragraph. It saves you the trouble of having to delete old text after typing new text.

Pressing ENTER in Typeover mode moves the insertion point down to the start of the next line.

Pressing CTRL+S in Insert mode splits a line at the insertion point. The text following that point becomes a new paragraph.

To join two paragraphs together, either position the insertion point at the start of the second paragraph and press the Backspace key, or position the insertion point anywhere on the last line of the first paragraph and press CTRL+A .

## Inserting New Lines

To insert new text in an existing paragraph, position the insertion point where you want the new text to appear and type the new text.

To insert a new paragraph, position the insertion point at the end of an existing paragraph, press ENTER and type the new text.

# Document Layout

The layout of a document refers to the way it is arranged on a page, in terms of the left margin, the right margin and the line length. When you create a new document, the Text Editor has default left and right margin settings of 5 and 75, giving a maximum line length of 70 characters. The left margin specifies the first position in which you can type a character at the beginning of a line, and the right margin specifies the first position in which you cannot type a character at the end of a line.

To control the document layout, the Text Editor uses a ruler line. You can set a ruler line for each paragraph in the document if you wish. All ruler settings are stored within Superbase SBT files.

## Showing the Ruler Settings

Initially, the ruler line is not visible. You can display it on screen by selecting the Ruler command from the Options menu. When you do this the ruler line appears at the top of the text window.

The angled bracket at the left of the ruler indicates the current position of the left margin, the angled bracket at the right of the ruler shows where the right margin is.

There are + characters along the ruler line at intervals of 8 characters. These indicate the positions where the next character may appear after TAB or SHIFT+TAB is pressed.

## Altering the Ruler Settings

You can move the left and right margin settings to any position on the ruler line. Place the mouse pointer on the ruler line at the point where you want to set a new margin and click the mouse button. If the pointer is nearer the left angled bracket, this action will reset the left margin; if it is nearer the right bracket it will reset the right margin.

When you alter the ruler settings, the Text Editor reformats the text of the document from the current paragraph onwards to fit the new margins. Margin codes are invisibly inserted in the text at the beginning of the current paragraph.

## Multiple Rulers

You can assign a separate ruler line to each paragraph. The ruler line at the top of the screen reflects the margin settings for the current paragraph. You can change the margins for a single paragraph by moving the insertion point to that paragraph and then resetting the ruler. But when you do this the paragraphs following the current paragraph may also be reformatted. The

point to notice is that reformatting applies to the current paragraph and any paragraphs following it which have not already had a ruler line set for them.

## Example

To illustrate the way multiple rulers work, let's consider an example involving a document with just four paragraphs. Initially, they all use the same ruler, the default ruler. Altering the margins for the first paragraph will have the effect of reformatting the entire document.

If you reset the left margin for the second paragraph to column 30, paragraphs 3 and 4 will also be shifted to column 30. The first paragraph, however, will stay at column 5, the default setting.

Next, move the insertion point to paragraph 3 and reset the ruler so that the left margin is at column 15. You will notice that this action reformats paragraphs 3 and 4 but leaves the other two paragraphs unchanged.

Finally, return the insertion point to the second paragraph and position the left margin at column 40. Because paragraphs 3 and 4 no longer use the same ruler, this operation affects only the second paragraph.

## Setting Margins

The Margins command on the Options menu supplements the ruler setting. When you select this command, Superbase presents a requester with three different margin control options: Current, Default, and External. For each option, you can set the left and right margin by typing its value in the Left and Right margin boxes.

- **Current** specifies the margins for the current paragraph; that is, the margin where the insertion point is located. Setting the margins with this options is equivalent to setting the ruler for current paragraph.

- **Default** specifies the margins for a new file. Use this option to set the margins before typing in new text.

- **External** specifies the margins that will apply when you load a non-Superbase document; for example, an ASCII or ANSI file. Files of this type do not contain any formatting information; but if you set the External margins, Superbase will format an external document when it is loaded into the Text Editor.

The margin settings are saved in the S:SUPERBASE.INI file when you click OK in the Margins requester.

## Text Styles

Four text styles are available from the Options menu: Plain, Bold, Underline and Italic. Styles are set by selecting the appropriate menu command.

# Search and Replace

## Search

To display the 'Search document' requester while running the Text Editor, select the Search option from the Search menu (or press AMIGA+? ). Specify the text string that you want Superbase to search for. If you want to find an exact match, click in the Match Upper/Lowercase checkbox.

When you click OK, Superbase searches forwards from the current insertion point, so if you want the whole file to be scanned you must position the cursor at the beginning of the file before you start the search.

The insertion point is positioned at the string if it exists in the document. If it does not, Superbase informs you and ends the search with the insertion point at the end of the document.

## Repeating a Search

To repeat a search for the most recently defined string, whether exact or not, select the Next option from the Search menu (or press AMIGA+> ).

## Replace

Replace allows you to replace every occurrence of a string of characters in the text with another string. You have the option of specifying an exact match for the string to be replaced, and you may also specify whether Superbase should confirm changes or not before replacing a string.

1.  Position the insertion point in the text before the first occurrence of the text you wish to replace.

2.  Select Search Replace (or press AMIGA+= ).

3.  In the box headed 'Search for', enter the text string you wish to be replaced.

    □  If you require an exact match, which takes into account the difference between upper and lower case letters, click in the Match Upper / Lowercase checkbox.

4.  In the 'Replace with' box, enter the string which is to replace the Search string.

    □  If you leave the Confirm changes checkbox turned on, Superbase checks with you before replacing each occurrence of the Search string. It first highlights the string and then displays a Replace Selection dialog which asks you for confirmation. Click Yes to proceed with the replacement, click No to skip it. Note that the dialog may be moved, allowing you to view the text.

    ❑  Click the Confirm changes checkbox if you do not wish to be
        prompted before a string is replaced.

5.  Click OK to proceed with the Search and Replace operation.

Notice that the Confirm changes checkbox also appears in the Replace
Selection requester. This means after any Search string has been found, you
can specify that the remaining strings be replaced without any further
confirmation.

# Control Key Equivalents

Instead of using the mouse for certain operations you may use Control key
equivalents. As well as those mentioned earlier in this chapter, the following
are available:

| | |
|---|---|
| Print | AMIGA+P |
| Plain | AMIGA+E |
| Bold | AMIGA+ B |
| Underlined | AMIGA+U |
| Italic | AMIGA+I |

# 24 PRINTING TEXT

Text documents are generally printed from within the Text Editor.

1. Check that the printer is connected, switched on, on-line, and loaded with paper.

2. Select the Text Editor from the Utilities menu.

3. Select Open from the Text Editor's Project menu.

4. Choose the document you wish to print by clicking on its name.

   □ If the document does not appear in the list, edit the filename extension in the pattern matching string. Type in '*.*' to view all the documents in the current directory. Superbase will load the file provided that it does not contain control characters.

5. Click on OK to open the document.

6. Select Print from the Text Editor's Project menu.

   □ You can interrupt printing by pressing CTRL+C .

The layout of each paragraph of text is determined in the Text Editor by the left and right margin settings. Superbase sends those lines of text to the printer unaltered, so it is important that the Set Printer Setup command has been used to ensure that the printer page width, margin settings and selected font combine to allow sufficient characters to be printed on each line.

## Further Information

Mail Merge operations are performed from within the Database, not the Text Editor. See Chapter 25 Mail Merge.

To print a program, use the Print command on the Project menu, and set the pattern matching string to '*.SBP' so that the requester displays the names of all your programs.

See also Chapter 13 Printing Data and Forms, and Chapter 23 Editing Text.

# 25 MAIL MERGE

With the Mail Merge command you can generate a large number of
personalized letters from a single Text Editor document. Each letter will be
based on the same 'form' letter that you have created in the Text Editor, but
the names and addresses will be inserted from a Superbase file which holds a
mailing list.

## Preparing for Mail Merge

Two things are required before you can print out letters with Mail Merge.
First, you need a file containing the names and addresses of all the people who
are to receive a letter – the targets of your mailing. Second, you need to create
the form letter which will be merged with these names and addresses.

## Creating a Form Letter

You create a form letter using the Superbase Text Editor. In appearance, it
will be like any other letter except instead of actual names and addresses it will
contain field names.

1.  Open the Superbase file containing names and addresses.

2.  Select the Text Editor from the Utilities menu (or press AMIGA+H ).

3.  Type the text of the form letter in the normal way.

4.  When you want to include a field name from the database file, position
    the cursor and then select the Merge Field command from the Edit menu.

5.  Superbase presents the Merge Fields requester with a list of field names
    from the current file. Click the name of the required field.

7.  Click OK.

8.  Superbase inserts the field name, delimited by the ampersand '&', into the
    text.

9.  Save the letter as a Text Editor document.

## Inserting Date and/or Time

You can insert the system date or time in a document. This is done by typing
the TODAY or NOW keywords delimited by ampersands directly in the text:

&TODAY&

## Using the & Character in the Text

If you need to use the ampersand character for other purposes, you should
enter it in the text as a double ampersand (&&). Superbase recognizes a

double ampersand in a text document as an intended single ampersand, so that text such as 'Laurel & Hardy' does not confuse the merging process.

## Fixed Length Fields

You may want to lay out a form letter in such a way that the contents of a field always start in the same column on the page. Normally, if a letter has two fields on the same line, the second field may be printed in a different position in each letter, depending on the contents of the first field.

The fixed length command lets you set the length of the first field; or, more precisely, it lets you fix the length of the output from the first field. By doing this, you can ensure that the contents of the second field are always printed at the same position. If the contents of the first field are shorter than the specified length, Superbase inserts spaces (if they are longer, it truncates them).

You set a fixed length for a field manually by typing dashes between the end of the field name and the second ampersand. The length is specified by the number of characters from the first ampersand to the second ampersand.

For example:

&Firstname---& &Lastname&

The fixed length for the Firstname field is 14 characters – nine characters in 'Firstname' plus two ampersands and three dashes. If the contents of this field are printed at column 1, the Lastname field will always be printed at column 16, no matter how long or short the first name is.

## Running Mail Merge

Mail Merge takes data from the records specified by a filter. It starts with the first record in the current index sequence. It replaces the field names in the form letter with the data from the corresponding fields in the record. It prints the letter and then fetches the next record. This process is repeated one letter for each record until the last record has been merged with the form letter. The procedure for printing letters using Mail Merge is as follows:

1.  Switch on and set up the printer.

2.  Open the database file containing the data which is to be inserted in the form letter.

    □  If you know that no text file is open, you can omit the next two steps.

3.  Select Text Editor from the Utilities menu and then open the text file containing the form letter.

4.  Return to the database by clicking in the database window.

5.  Select Mail Merge from the Process menu.

  ❑ If there is no text file open, Superbase presents you with a requester containing a list of text files. Click on the name of the required form letter, then on OK.

6. You will be presented with a Filter requester. Set any required filter conditions to select records for merging (see Setting the Mail Merge Filter below).

7. Click OK. Another requester will appear:
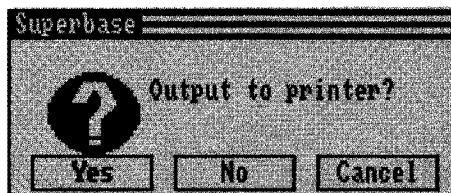


8. Click Yes to start the Mail Merge operation.

## Setting the Mail Merge Filter

You can use the Mail Merge filter to specify which records are to be used. If you leave the filter line blank, Mail Merge will print a letter for each record in the file.

Mail Merge presents you with the standard filter requester, and you build up your filter command line in the usual way. Here are some examples of a Mail Merge filter line:

  Date > "1 September 1990" AND Date < "31 August 1991"

This line selects all the records where the date field falls between these two dates.

  Lastname LIKE "S*"

This filter instructs Mail Merge to print letters for all the people in the file whose names begin with S.

## Previewing Mail Merge

Mail Merge allows you to preview letters on screen before committing them to print. This enables you to see what the letters will look like when the names and addresses have been inserted in them.

The preview option is selected at the stage when the Output to Printer requester appears. Clicking No instead of Yes causes Mail Merge to output letters to the screen.

Preview displays each letter in turn, from the first letter onwards. If you set Paging on (from the Set menu), the preview pauses whenever the screen becomes full.

Click the Pause button or press SPACEBAR when you want to move on to the next letter.

Click the Stop button or press CTRL+C to bring the process to a halt. Otherwise, Mail Merge will preview all the letters that have been selected by the filter. If you intend to print a large number of letters, previewing all of them may take some time.

Notice that letters are reformatted after the record data has been inserted. This means that the contents of a field may move the text which is on the right onto to the next line. As a result, text on the last line of the page will be moved to the start of the next page.

## Blank Lines

One of the problems with merging a set of address fields is that in some records one of the fields may be empty. For instance, the address may not include a city name. If this line were left blank when the letter was printed, there would be an unsightly gap between the street name and the zip code.

Superbase solves the problem by checking whether a field is empty or not. If it is empty, the field is not merged and the fields above and below are printed on consecutive lines.

## Mail Merge Applications

Mail Merge does not restrict you to names and addresses. This is simply the most common mail merge application. Any field name may be used and it may be placed anywhere in a document. You could create a form letter which said:

> We would like to remind you that payment on the invoice dated &Date&, for the sum of &Amount&, is now 60 days overdue.

In this case, though, the fields **Amount** and **Date** would have to be present in the same file as the name and address fields. It is possible to use field names from more than one file, provided the file name is given as well as the field name. However, when the letters are printed, Superbase will only select records from the current file. The current record in another file will stay the same and a field from this file will print the same data in each letter.

Even so, there are some circumstances in which Mail Merge's multi-file capability can be useful. Suppose your company was sending out letters in batches, where each batch included all the customers in a particular area. By using a **Dealers** files you could give each customer information about the dealer in their area. The form letter would include the line:

> The dealer for your area is &Name.Dealers&, &Address.Dealers&

To do this, you would need to select the required record from the **Dealers** file before printing each batch of letters.

Fields may also be repeated in a letter.

## Mail Merge with Multiple Files

The Mail Merge command works with a single file. If you require to merge data from more than one file into a document, all you have to do is create a new database file containing the joined data with the Query to SB File command. See Chapter 16 Reorganizing the Database. You may then open the new file and select Mail Merge.

Alternatively, you may prefer to program the application as illustrated below.

## Programming Mail Merge

If your mail merge application is more complex than usual, perhaps because it involves extracting data from multiple files, you will need to construct a simple DML program to solve your problem.

- You may place variables in a mail merge document.
- You may use the ? TEXT MERGE command to output a merged document under program control.

First, change your merge document. Place any variables you wish to use inside ampersands: &x$&.

Second, write a program to assign the required data to the variable after loading the text document, and then print it:

```
OPEN FILE "Dealers" : SELECT FIRST
LOAD TEXT "LETTER1"
WHILE NOT EOF ("Dealers")
    X$ =  credit_limit.Dealers
    ? TEXT MERGE
    SELECT NEXT
WEND
```

- Include the PRINT command before the WHILE line to direct output to the printer.

You must ensure that your program assigns a value to every variable before you try to merge it, or Superbase will display the 'Variable Not Defined' error message.
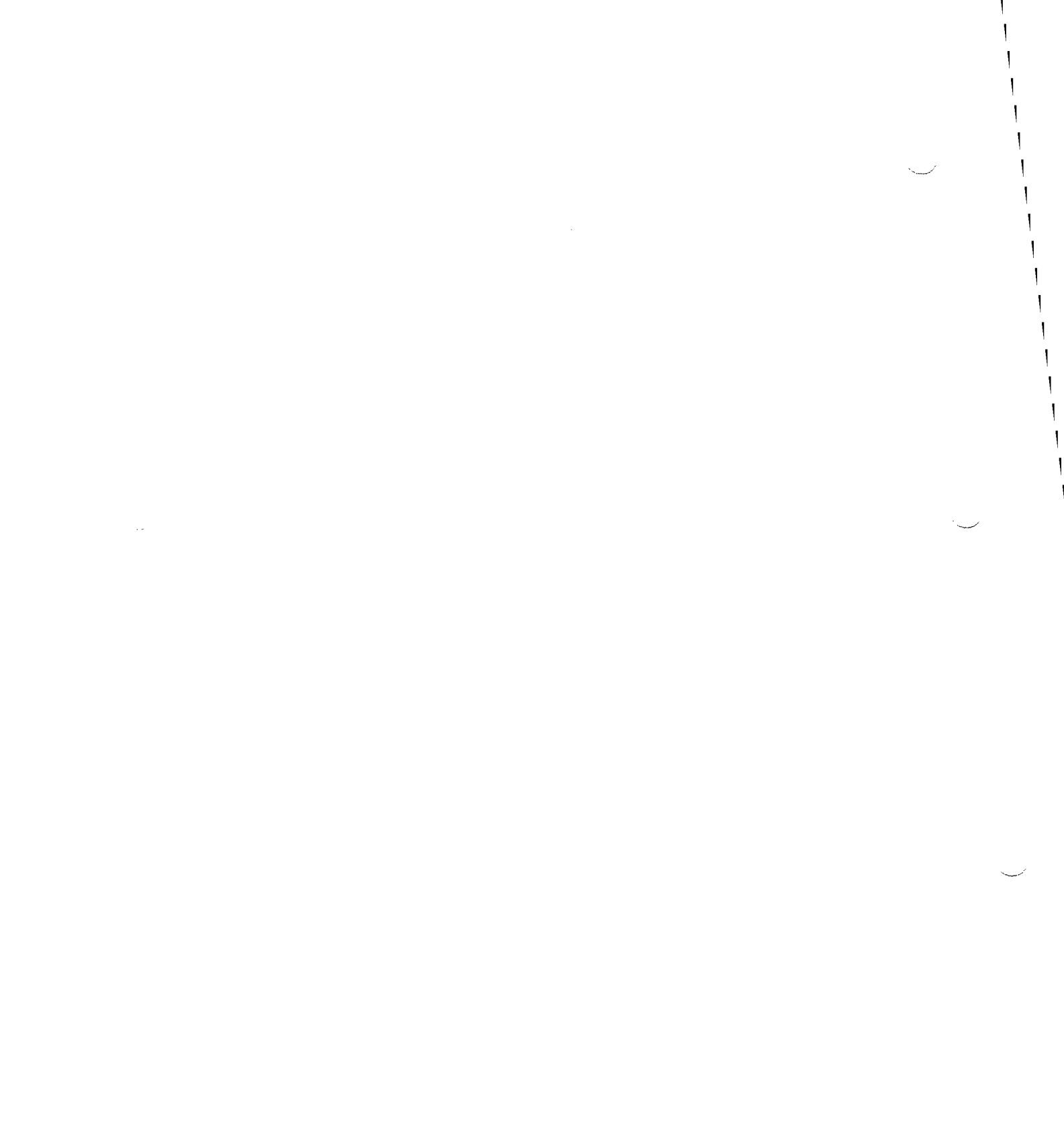
# 26 LABEL PRINTING

You can print multiple mailing labels – as many as fit on the page – using the Process Labels command. You can control the dimensions of the label, as well as the number of fields per line. Labels formats are stored in Superbase SBB disk files.

Labels can only be printed from the current file.

1.  To create a new labels file, or to continue editing one you already have in memory, select Labels Edit from the Process Menu. (Labels Open and Labels Save As options are described later in this chapter.)
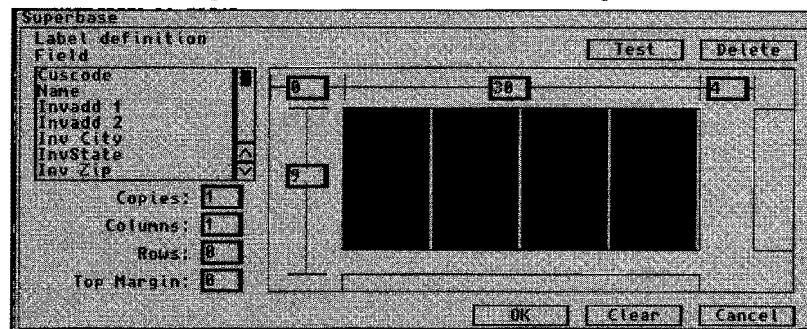


2.  Design the field layout of your label. Highlight the position you want a field to appear in, then click in the Fields list to copy the name of the field into the layout.

    □  Fields always appear on a line from left to right without gaps.

    □  To delete a field from the layout, highlight it and then click Delete.

    □  The selection of fields for the labels layout overrides the current Field Selection.

3.  Define the horizontal dimensions of the label. The three boxes at the top of the panel set the left margin, the width of the text to be printed on each label, and the gap to the next label along. Adjust these values to fit your label stationery. Dimensions are measured in characters.

4.  Define the vertical dimensions of the label. The box at the left of the label layout sets the number of lines from the top of one label to the top of the next.

5.  Set the number of copies per label, up to a maximum of 99.

6.  Set the number of columns of labels, up to a maximum of 8.

7.  Set the number of rows of labels to be printed per page. Enter zero if the feature is irrelevant.

8.  Set the number of lines to offset vertically before starting to print.

9.  If you want a test print to check whether you have set up the dimensions correctly, click Test. Superbase prints one page of labels, using the character 'X' to show the width of the text on each label.

10. Click OK to proceed to the Filter requester.

11. Enter a Filter Command Line if you want to select records for printing, or leave the filter blank if you want to print labels for all the records in the file.

## Storing Label Formats on Disk

Superbase's label formats are re-usable. Label formats are stored in files with a SBB extension. These files contain information on field layout, dimensions, and other parameters. To store the current format on disk:

1.  Select Labels Save As from the Process menu.

2.  Enter a filename for the format file that will be created.

3.  Click OK.

When you want to use an existing labels format, you follow a similar procedure:

1.  Select Labels Open from the Process menu.

2.  Select a filename from the list (changing directory if necessary).

3.  Click OK.

    ❑ If the selected format uses a file other than the current file, Superbase opens the file and makes it current.

    ❑ The set of fields defined in the SBB file overrides the current selected fields list.

## Further Information

When printing, Superbase automatically closes up lines if all fields on a line are empty. A space is automatically inserted between fields on the same line.

If your label print run fails before it has finished, perhaps because the stationery jams in the printer, you can use Superbase's Filter to restart from a point in the file near where the failure occurred. Enter a Filter Command Line that states keyfield > "value", where keyfield is the current index, and 'value' is the value of the indexed field in the last successfully printed record.

# 27 COMMUNICATIONS

Superbase's Communications facility allows you to transfer files from one computer to another. In most cases both computers will be running Superbase, but you can also send files to other communications packages which support one of the Communications protocols.

## Local or Remote
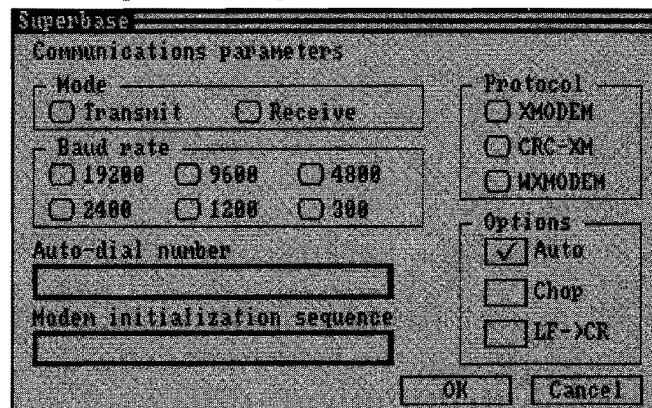
There are two ways of connecting computers for file transfer. A local transfer is one where the computers are in close proximity (usually, in the same room) and are directly linked by a cable which is connected to their serial (RS-232) ports.

With remote transfer, files can be sent over almost any distance via the public telephone networks. Both computers use a modem connected to a telephone or telephone socket.

If the modems have an auto-dial and auto-answer facility, you can establish the link between them from within the Communications requesters. Otherwise, you must establish the link before selecting the Communications command.

## Using Utilities Communications

Select Communications from the Utilities menu. Superbase displays the communications parameters requester.



Before a file can be transferred, both users must set up all the parameters in this requester in the same way.

For each of the following parameters you must select an option by clicking on its corresponding button:

**Baud Rate**
This determines the speed at which the file data is transmitted. Six baud rates are available: 19200, 9600, 4800, 2400, 1200 and 300.

**Protocol**
Three transfer protocols are supported; XMODEM, XMODEM-CRC and WXMODEM. If you select WXMODEM and/or XMODEM-CRC for transfer, Superbase will use it if possible, but is also capable of detecting whether the receiver is also using WXMODEM, and of 'changing gear' by automatically selecting first XMODEM-CRC and then XMODEM if necessary. This means that in practice the receiver determines the protocol to be used.

**Options**
**Chop** removes the padding from the last block of a received file. It removes zeros, nulls, and CTRL+Z  characters. XMODEM always sends files in 128 byte blocks. If the last block is not complete, it is padded out.

**LF-CR** works on transmitted files only, converting Amiga Linefeed characters (ASCII 10) to Carriage Returns (ASCII 13), enabling text files created on an Amiga to be read by editors on PCs and other machines.

**Auto** is only used when you are transmitting files between two computers which are both are running Superbase. If Auto is selected, Superbase transmits the file's name and the file is saved under this name at the receiving end. If it is not selected, the receiver specifies the name under which the file is to be saved. Superbase displays a file selection requester before transmission takes place. If you are at the receiving end of the Communications link, you must then enter a name for the file or select an existing name from the requester panel. The header block will always be transmitted using XMODEM, regardless of the protocol selected for the actual file transfer.

**Transmit/Receive**
Click the Transmit button if you are sending a file, click Receive to receive a file.

**Auto-dial Number**
If your modem has an auto-dial option, you may enter the receiver's number in this box. Start this number with a 'T' to specify tone dialling or 'P' to specify pulse dialling. If you specify neither, Superbase uses the modem's default mode. Superbase automatically sends the string 'ATD' before the number.

**Modem Initialization Sequence**
You may insert a sequence of up to 28 characters, which will be sent to the modem when you click OK, before file transfer begins.

**Note**      All the parameters that are set with Communications requesters are stored in the S:SUPERBASE.INI file when you click OK.

## Wild Card

You may use the '*' character to represent either a file extension or a filename. The #? syntax of AmigaDOS is not supported.

## Transferring a File

When both users have set the parameters in the Communications requester, they should click OK. If you have selected Receive, the computer will then wait for a file to be transferred from the other end of the link.

If you are transferring a file, you will be presented with a file selection requester for the files in the current directory. Select the file you wish to send and click OK. Superbase will then proceed to transmit the file.

During transmission, a further requester appears on both machines showing the status of the transfer. It displays the number of blocks successfully transmitted or received, and indicates errors. If errors occur, the block is transmitted again.

When transfer has been completed, the file is stored on disk at the receiving end. If Auto has been selected, it is stored under the same name with which it was transmitted.

Note: If you are transmitting files over the public telephone network at a baud rate higher than 1200, you may experience problems with file transfer. In order to reduce the number of errors, you should select a lower baud rate.

## The Transfer Status requester

During file transfer, Superbase displays a requester showing the status of the transmission. The top of the requester indicates whether the user is transmitting or receiving and, if the Auto has been selected, it gives the file name. It also shows the number of blocks to be transmitted and the type of transmission mode selected.

Below this, there are four headings: Count, %Complete, Errors and Consecutive Errors. Count gives the number of blocks that have been successfully transmitted; %Complete shows this number as a percentage of the total number of blocks; if there are any errors in transmitting file blocks, they are indicated by Errors and Consecutive Errors.

## Remote Transmission

The procedure for sending files via modem is as follows:

1.  If Auto-dial is required, enter the receiver's number in the Auto-dial box in the Communications requester.

2.  Select Transmit.

3.  Set the other Communications parameters for baud rate, port number, etc.

4.  Click OK.

5.  Select the file or files (using wild cards) for transmission.

Notice that there is no Remote/Local option in the Communications requester. Provided the modem is on line and the DCD line is off, Superbase detects that Remote transmission is required and proceeds with the operation as soon as you have selected a file.

At the receiving end:

1.  Select Receive and then set up the other parameters so that they have the same values as in the transmitter's Communications requester (although the port number does not need to be the same).

    □ If your modem has an autoanswer facility, this must be turned on.

    □ If it doesn't have autoanswer, the ring indicator line from the modem must be connected to the computer's RS 232 port.

2.  Click OK in the Communications parameters requester.

    □ If the Auto has been set to Off, you will first be presented with the file selection requester where you can enter the name under which the file is to be saved. Superbase will then display the 'busy' symbol (a watch icon) and will wait to be contacted by the transmitter.

    □ If the Auto option has been selected, transmission will proceed as soon as the communications link has been established.

ATD and ATA can be used in the modem initialization sequence to establish file transfer during an existing telephone call. ATD should be entered by the transmitter, ATA by the receiver. This means that the transmitter will not need to enter anything in the Auto-dial number box.

## Cable Connections

If you are using a modem, then you should use the cable supplied with it. Superbase uses the DCD line from the modem (pin 8) to determine whether a connection has been established, so make sure this pin is connected. Also, if your modem has options which change the usage of the DCD line, make sure that you select the option that causes DCD to reflect the current connection status.

If you are connecting two computers by a direct RS232 line, they should be wired up as follows:

| | 25-pin 'D' | | 25-pin 'D | 9-pin 'D' |
|---|---|---|---|---|
| XMIT | 2 | ------\/------ | 2 | 3 |
| RCV | 3 | ------/\------ | 3 | 2 |
| RTS | 4 | ------\/------ | 4 | 7 |
| CTS | 5 | ------/\------ | 5 | 8 |
| DSR | ┌─ 6 | | 6 ┐ | 6 ┐ |
| GND | │ 7 | ---------------- | 7 │ | 5 │ |
| DCD | └─ 8 | ------\/------ | 8 ┘ | 1 ┘ |
| DTR | 20 | ------/\------ | 20 | 4 |
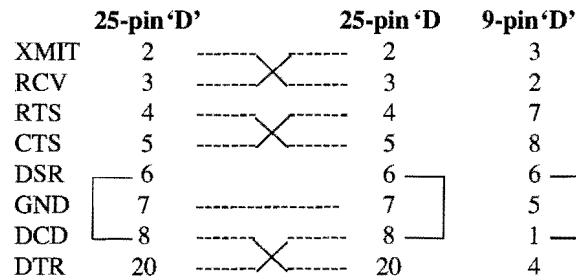
## Communications under DML

Superbase's communications functions are also accessible from the DML programming language. Using DML, you may write programs for keyboard communication with another Superbase user, as well as for unattended file transfer.

The DML commands used for communications are:

        CLOSE  COMMS
        COMMS  ?
        COMMS  GET
        COMMS  INPUT
        COMMS  FILE
        COMMS  FILE  GET
        OPEN  COMMS
        WAIT  COMMS

Further details are available from the descriptions of individual commands in the DML Reference Guide.

# 28 INTEGRATING TEXT AND IMAGE FILES

Superbase's external file management system allows you to store the names of external files in a Superbase database file, and display both the database records and the external files themselves. There are three types of external file: text files, image files and sound files.

## Text Files

External text files provide a bulk text or memo facility. They are used to store the same kind of information as database text fields but do not carry the same limitations on the amount of information. The maximum length for a text field is 4000 characters. The maximum size of a text file depends on the amount of free memory available. If you try to load a document for which you have insufficient memory, Superbase loads as much of the document as it can and displays an error message.

## Image Files

Image files hold graphic images. Usually they will have been created in a painting or drawing program, but you may also use them for digitized images. The restriction here is that the digitized image must be stored in a format that Superbase accepts.

## Sound Files

Sound files typically hold digitized sound samples, which can be played back. IFF (8SVX) format or plain sampled sound are supported.

# External File Field

In the File Definition, there is an External File field type. External File fields are like Text fields, except that Superbase can use their contents as path names for external files. A path name is simply a path through the disk directory structure to the file itself. The contents of an external file field can be:

- A filename.
- A path specification followed by a filename.
- A drive name followed by a path specification and filename.

These three ways of specifying an external file are described in turn below.

When you enter the name of an external file, you must supply the full file name, including any extension. For example, if you are using a text file which you have created in the Text Editor, you must supply the SBT extension:

Comment.sbt

### File Name (same disk, same directory)

At its simplest, an external file field refers to a picture or a document in the same directory as the database file, for example:

Roomplan.pcx

This is the name given to a picture created with a paint program.

### File Name (same disk, different directory)

If you want to access a file in another directory you must include the name of that directory in the external file field. Superbase can then go to that directory, open the file and read the data.

Suppose you kept picture files in a directory called Images. The external file field contents would have to look like this:

:Images/Roomplan.pcx

### File Name (different disk)

If your external file was on another disk, you would have to start your path name with the disk drive specifier, for example df0: or dh0:. Continuing with the text file example, suppose you kept a text file in a directory called Documents on the disk in drive df0:. The external file field would have to hold the following path name:

df0:Documents/Comment.sbt

The colon tells Superbase that the Documents directory is to be found on the disk in drive df0:. Superbase will therefore expect to find a disk in that drive. If it does not find a disk, or does not find the directory, Superbase will display an appropriate error message.

**Note**      Although you can specify a path to be relative to the current directory, you should always include a full pathname if you may be changing directory during a session in which external files are being read.

# External File Window

### Text

External text files are displayed in the Text Editor window. When you click the External File button, Superbase loads the file into the Text Editor and makes the Text window active.

At this point, you can treat the file in the same way as any other text file. Clicking the External File button has the same effect as selecting the Text Editor in the Utilities menu, and then loading the file with the Open command in the Text Editor File menu.

In most cases, you will only want to view the file; if it is longer than the text window, use the scroll bars to scroll it into view. But you can also take advantage of any of the Text Editor functions. Thus you could insert new text and save the file on disk, or you could replace the text with another document, and then save it.

## Text Types Supported

Superbase accepts two types of text file.

*   SBT files created in the Superbase Text Editor.
*   ASCII files created in another text editor or in a wordprocessor.

## Graphics

Graphics images are displayed on a separate screen.

You can manipulate the Graphics screen using the standard Workbench controls.

To close the Graphics screen, display the Graphics screen's Project menu and select the Close command.

## Image Types Supported

The compatible image types are:

IFF (normal Amiga images)
PCX (PC Paintbrush files)
GIF (Compuserve Graphics Interchange format)

The supported types of IFF files are ILBM, Dynamic Hi-res and Dynamic HAM.

**Note**    The names of any PCX or GIF files must end with '.PCX' or '.GIF' respectively for Superbase to recognize them.

# Viewing External Files

Text, graphics and sound External Files are viewed in the same way:

*   Select for browsing in the usual way a Superbase data file which has at least one External File field in its file definition, and which has the required external files named in some of its records. At first, browsing through the records of this file simply displays the contents of its records and does not display the contents of any specified external files.

- Click on the External File button – the camera gadget at the right of the control panel at the bottom of the screen – or press the keyboard equivalent, which is the '*' key. As long as this button remains selected, each time you select a new record any external file named in it is displayed in the appropriate window.

- When the contents of an external file are being displayed, a set of three menus is available. Press the right-hand mouse button to examine these menus.

- If a record contains more than one external file field, the files for these fields can be thought of as being on different levels. The Next External and Previous External commands on the main Record menu are used to move between different external file 'levels' within a single record. Click Next External to move to the next file down; click Previous External to move back to the previous level. However, you do not need to return to the main Record menu to use these commands since they are duplicated in the local Record menu, available when an external file is being displayed.

- The local Field menu, available when an external file is being displayed, allows you to move from one image to another if the external file you are displaying contains more than one image.

# Image Scaling and Aspect Ratio

External file images are normally loaded and displayed without modification. Color images appear in color, images are not scaled in size, and aspect ratio is not preserved.

However, it is possible to apply specific rules to the display of individual external images by appending parameters to the name of the image file to force a gray scale display, preserve aspect ratio, and scale the image to fit its box or window. Gray scale display is covered in the next section, and default values and priorities of parameters are described later in this chapter.

## Image Scaling

In this context, image scaling refers to the way in which the size of an image is reduced or increased as it is loaded from disk.

An image will always be scaled to fit its form box if the 's' parameter is added to the filename:

"eagle.pcx,s"

- Note that the parameters must form part of the filename literal – they fall within the quotation marks.

The Form Designer includes an Image Scale tool. This allows you to apply a display attribute to a form image (graphic as opposed to data). When applied

to an external file field, Image Scale affects *all* images displayed in the field, and is equivalent to the 's' parameter.

## Aspect Ratio

Aspect ratio refers to the fact that the vertical and horizontal dimensions of any given image have only one correct relationship. If an image is distorted either vertically or horizontally, its aspect ratio is lost. In some applications this may not matter, but in others the aspect ratio must be preserved.

An image will always preserve its aspect ratio, avoiding any distortion that might occur if the form box has different proportions from the image itself, if the 'a' parameter is used:

"eagle.pcx,a"

If the 's' and the 'a' parameters are used together the 'a' parameter takes priority. Also, the Image Scale attribute is overridden by the 'a' parameter.

# Gray scales and Color Mapping

Many image intensive applications involve images that are scanned in 64 or 256 shades of gray, or 256 colors. This exceeds the number of colors that the Amiga can display without resorting to specialist hardware and software.

Superbase displays such images by color mapping, which involves analysing the colors used in the image and 're-mapping' them to the closest color available in the screen's palette. The resulting image has fewer colors than the original, but is generally recognizable. The quality of the displayed image depends somewhat on the palette of the form, or external file screen, to which it is mapped. Superbase offers three options:

- **Map to Default**

  "eagle.pcx,f"

  This maps the image colors to the current color palette on a form, or to a default palette on an external file screen. The default form palette gives good results with many images and is useful (in conjunction with the scaling options) for displaying together images that have widely differing palettes.

- **Map to Color Palette of Image**

  "eagle.pcx,c"

  This forces the color palette of the form or screen to be the same as that of the image. Colors are read from the image in the order they occur, up to the maximum number of colors that can be displayed (16 for a form, or up to 32 on the external file screen). Each pixel in the image is then mapped to the closest color of those available. Images that benefit most from this option are those with 'prioritised' color palettes, in which the

most important colors appear first in the list. Since you often have no control over the color palette of an image, this option may cause menus and control panels to be rendered in similar colors, making them hard to read.

•   **Map to Gray Scale**

"eagle.pcx,g"
This sets up a gray scale and displays the image in shades of gray. This option can often be the most satisfactory, even without color, especially with 256-color images.

The relative priorities of these parameters are described later in this chapter.

If none of these parameters is specified, Superbase attempts to map to the full image color palette, otherwise it maps to a default.

# Image Justification

If a displayed image is smaller than the screen, you may position it in the top left corner, the center, or the bottom right corner of the screen. If you use the same parameters to display an image that is larger than the screen, you will cause the top left, the center, or the bottom right sections of the image to be displayed.

"eagle.pcx,["

gives 'top left' justification,

"eagle.pcx,|"

gives 'center' justification, and

"eagle.pcx,]"

gives 'bottom right' justification.

If none of these parameters is specified, Superbase centers an image on the external file screen, or left justifies an image on a form.

# Parameter Priorities

The eight parameters described above (s, a, f, c, g, [, | and ]) can be specified singly or in any combination and in any order, with or without separating commas. For example:

"eagle.pcx,s,c"
"eagle.pcx,cs"

The various parameters can, however, interact with one another, so have priorities associated with them. In any case of conflict, the parameter with the

higher priority takes precedence. If you do not specify a parameter in one or more of the groups (scaling, mapping or justification), then default values are assumed.

Priorities are summarized in the following list, in which the parameters appear in order of priority, starting with the highest.

| | |
|---|---|
| s | Scale to fit |
| a | Aspect ratio preserved |
| f | Form palette to be used |
| c | Color palette of image to be used |
| g | Gray scale display |
| [ | Top left justification |
| | | Center justification |
| ] | Bottom right justification |

# Images in the Form Designer

When loading an image as part of a form design, you may choose whether to use the image palette, the existing form palette, or a gray scale. If the image palette is used, the Form Designer redefines the form palette with the image's colors. If a gray scale is used, the Form Designer redefines the form palette to a gray scale.

The image selection requester also includes options to scale an image while loading it. The options have the same effect as the 's', 'a', 'g', 'f' and 'c' parameters described in earlier sections.

The Image Scale tool affects only the display of images, and is relevant mainly to images displayed as data in Superbase.

# External File Fields on Forms

You may place one or more external file fields on a form, so that data may be seamlessly integrated with images and text. However, you should be aware that such an application, like all graphics applications, must necessarily be memory intensive, and you may need to fit additional graphics memory if you plan to concentrate on this area.

The external file field on the form may occupy an area defined as larger or smaller than its real size. There are provisions within the Form Designer for making such a field clip or scale its contents. Likewise, you may choose the justification of an image within its frame on the form.

To view all the external files referred to on a form, click the camera button.

### Editing External Text Files on Forms

You can invoke the Text Editor window simply by clicking within the displayed area of text on a form. Superbase creates a window the same size as the external file field. You may perform any normal editing activity, including saving the document.

Selecting the next record causes the Text Editor window to close.

### Printing External Files

External files, both images and text, may only be printed as part of a form.

1.  Define a form with external file fields positioned as you require.

2.  Open the form in Superbase.

3.  When the images and/or external text are displayed, print the form in the normal way.

See Chapter 13 Printing Data and Forms for further information.

# Querying External Files

You can use the facilities of the Query command on the Process Menu to produce text or picture output in conjunction with data output. Two steps are necessary to achieve this:

1.  Select the External File Query button in the System Options requester on the Set Menu.

2.  Name the External File field or fields that you want to view in the Query Fields line.

This feature applies to the Query command on the Process menu. If External File Query is OFF and you output an External File Field as part of a Query, you'll see only the textual content of the field: the name or pathname of the external file. However, if you set this feature ON and then output an External File Field in a Query, you'll see the contents of the external file, as if you were doing a Fast Forward.

External Files are not printed if the Query output is directed to the printer.

# Searching External Text Files

Superbase provides a special operator for use with external text files: the CONTAINS operator. CONTAINS works in the same way as the LIKE operator, but whereas LIKE is used to search for text strings in text fields, CONTAINS searches for text strings in external text files.

CONTAINS can be entered in any Superbase filter command line, but it will be used most commonly with the browsing controls filter or with the Query filter. It must be preceded by the name of an external text field, e.g.:

Main_Entry CONTAINS "*tractor*"

In this example, the name of the external text field is **Main_Entry**. In each record in the database file, the **Main_Entry** field stores the name of an external text file. When you enter this line in a filter, Superbase searches all the external text files associated with the **Main_Entry** field for an occurrence of the word 'tractor'. If it finds the word, Superbase displays the text file in the Text Editor window.

You will notice that the example filter line makes use of the pattern matching character '*'. The reason for this is that CONTAINS searches for matching lines rather than matching words. This means that you need to define the search string as an entire line rather than a single word. The word 'tractor' may be located anywhere on a line, so you need to take this into account by placing the '*' character before and after the word in the search string. If you were sure that the word only occurred at the beginning of a line, you could omit the first '*' character, as in:

Main_Entry CONTAINS "tractor*"

You may also use any of the other pattern matching characters that are available with the LIKE operator. For example:

Main_Entry CONTAINS "*[a-d]. Smith*"

This will search for all the people whose last name is Smith and whose first name initial falls in the range A to D, such as 'A. Smith' and 'D. Smith'.

Because CONTAINS searches on a line by line basis, it is suitable for use with structured text files, such as some ASCII file output from other applications, rather than with free format text.

# 29 FUNCTION KEYS

The action of function keys may be defined by the user. You may attach a command line of up to 255 characters to any of forty function keys. Pressing the key to which the command line is attached executes the command line.

Function key definitions may be saved as SBK files and are re-usable.

The main applications for this facility are:
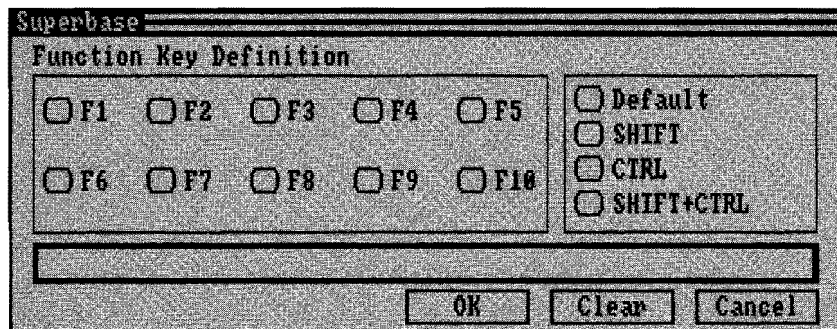
*   Creating command lines to perform multiple operations with a single keystroke.

*   Inserting often used words, keywords, phrases, and paragraphs in a text document or a program.

There are forty function keys available by combining F1 to F10 with control keys in four different ways:

> Function Key alone
> SHIFT+ Function Key
> CTRL+ Function Key
> SHIFT+CTRL+ Function Key

## Defining a Function Key

Select the Function Keys command from the Set menu. You will then be presented with the Function Key Definition requester:



First, choose the key to which the command line is to be attached.

1. Click the radio button for the required combination in the box at the right-hand side of the requester.

2. Click the radio button for the key itself.

3. Type the command line to be executed in the text box. Do not press ENTER at the end of the command line unless you wish to exit from the requester immediately.

   ☐ Multiple commands may be entered, separated by the colon character ':' as in a DML command line.

   ☐ You may specify text on its own. Pressing the key prints the text in a field or in the Text or Program editor.

   ☐ If you want to be able to execute the command line at any time, for example in the middle of data entry or while editing a program, type an exclamation point '!' as the first character. In data entry, this has the effect of pressing the ESC key and ends the activity.

4. Click OK to exit from the requester.

The function key is ready to use.

### List Function Key Assignments

Select the Function Keys List command from the Set menu. Superbase displays the current definitions in the work area.

**Note** The DML KEY command may be used to list output. You can redirect the output to the printer with a DML Command Line:

    PRINT : KEY : DISPLAY

# Function Keys for Commands

The rules governing the commands which can be assigned to function keys are the same as for the DML Command Line. If the commands can be executed as a one line program, they can also be assigned to a function key.

**Note** Single word commands at the beginning of a function key string must have a trailing space if followed by a colon.

### Examples

The following function key string simply displays the current time, but it also illustrates the use of the '!' character. Placing this character at the front of the string forces Superbase to execute the command in all circumstances:

    !? NOW

The second example could be adapted to any situation where you wish to select a record but are not sure what to enter as the index key. In this case, it allows you to enter just the initial letter of a customer name (or the first part of the name) and then displays a list of the names in the customer file that begin with the letter (or letters) you have entered.

```
a$ = "":REQUEST "Enter customer initial","",4,a%,a$,15:REQUEST
"Selectcustomer","",20,a%,a$,40,Lastname,Firstname:SELECTKEYa$
```

The last example has a more specialized application. Suppose you are entering details of check payments in a file called **Checks**. For each check, you create a new record which holds details such as the amount and the date. You also want to keep a running total of the amounts as they are entered. One way of doing this would be to define the following function key string:

```
SELECT LAST:x% = Total.checks: BLANK: ENTER: Total.Checks =
Amount.Checks + x%: STORE: VIEW
```

Pressing the function key for this string collects the total from the last cheque entered and then presents you with a blank record. When you have typed in the details of the next cheque, it updates the **Total.Checks** field and stores the new record.

The example assumes that the file is currently indexed on a date field and that the cheques are entered in date order.

# Function Keys for Text and Program Editing

Function keys are useful for entering text strings which occur many times in the same document or are regularly used in different documents. They save you the trouble of having to type in the string yourself every time it is needed. Provided the text insertion point is shown on screen, pressing a function key will insert a string at that point.

The same applies to the use of function keys when editing a program. You can define function keys to insert commonly used keywords in a program.

For example, you might want to define the keys SHIFT+F1 to SHIFT+F5 as:

```
REM
LOAD"
SAVE"
OPEN FILE"
WHILE
```

Keys SHIFT+F5 to SHIFT+F10 could then be assigned commands which are useful during program development:

```
!MEMORY
!RUN
!PAGING ON:? LIST
!EDIT
!STATUS
```

Note the use of the '!' character here. It allows these function key commands to be executed even when the program insertion point is active.

If you wish to define a text string that runs over several lines, you need to use the carriage return character. However, it is not possible to enter this character in the Function Keys requester, so you must enter it using the KEY command in a program or a DML Command Line. So to assign an address to F1, you would enter a string like this:

```
KEY 1, "8404 Sterling Street" + CHR$(13) + "Irving" + CHR$(13) +
"Texas 75063" + CHR$(13)
```

**Note**      Superbase cannot save control characters in a function key file, so to store this string on disk you need to save it as a one line program.

Normally, function key strings which are intended for text entry cannot be executed as commands. Pressing a function key of this type when the cursor is not active may cause an error message.

# Editing a Function Key String

If a function key has already been defined, selecting the Function Keys command from the Set menu displays the contents of the key in the string entry box. Initially, the setting for the F1 key will be shown. When you choose another key with the requester controls, Superbase displays the setting for that key.

If you wish to change the current setting, you can do so using the usual editing controls.

Alternatively, click Clear to remove the current setting and type in a new function key definition.

# Function Key Files

Function Key definitions can be saved on disk for use in other sessions with Superbase. Function key files are saved as ASCII files with the SBK extension. If you wish to examine a function key file, you may display it on screen or print it out with the Type option on the Utilities menu. Any keys which have not been assigned a string are shown as blank lines.

## Saving Function Key Files

1. Select the Function Keys Save As command from the Set menu.

2. Enter a filename.

3. Click OK.

Superbase saves the current function key definitions on disk, adding the SBK extension to the file name you type in.

To save a function key file under program control or from the DML Command Line, use the command SAVE KEY "file name".

## Opening Function Key Files

1. Select the Function Keys Open command from the Set menu.

2. Choose a filename from the list.

3. Click OK.

Superbase loads the selected file into memory and sets up the function keys ready for use.

Use the LOAD KEY "file name" command to load a function key file under program control or from the DML Command Line.

You can include this command, followed by the name of a function key file, in Superbase's auto-executing **Start.sbp** program (see Volume 2, DML Reference Guide). This way, the function keys will be set up for you as soon as Superbase has loaded.

Another way of using the LOAD KEY command is to assign it to a function key. You will then be able to load a specified function key file with a single key press.

# 30 HOUSEKEEPING

Housekeeping tasks are those that lead to efficient file management: deleting files, renaming them, documenting them, and, most important of all, making backup copies of database files.

## Removing Database Files

Use Remove on the Project menu to remove a database file from disk. This command only operates on open files and erases all the files that make up a single Superbase database file: the SBF file, the SBD file, and any indexes associated with the file.

When you select the command, Superbase presents a standard file requester with a list of all open files. You select a file in the normal way:

1. Click its file name.

2. When the name you want is in the Text box, click OK.

Before finally removing the file, Superbase asks for confirmation that this is indeed what you wish to do.

   ❑ To be able to remove a file, you must have opened it with delete privileges. You automatically have these unless more than one password has been set.

**Notes**  Do not confuse Project Remove File with the Process Remove command. The latter is used to remove specific records from a file while leaving any remaining records and the file definition (the SBD file) intact.

Form files (SBV) may be deleted from within the Form Designer or with the Utilities Delete command.

## Removing Index Files

### Project Remove Index

From time to time you may wish to drop an index if you find it is rarely used. This will improve performance slightly and use less memory, as Superbase will no longer have to update the index when you add or replace a record.

As for Project Remove File, Superbase presents a standard requester, this time with a list of all the indexes for the current file. You select an index in the normal Superbase way:

1. Click its file name.

2.   When the index name you want is in the Selection box, click on OK.

☐   As this too is irreversible, Superbase asks for confirmation that this is indeed what you wish to do.

**Note**     To change a unique index to duplicate index or the other way round, first use this command to remove the index; then rebuild the index, using the Index New command.

# Backing up Data

To safeguard against accidental loss of data, it is important to make backup copies of your database files on a regular basis.

One way of doing this is to use the Copy command on the Utilities menu. Remember that a single database file has several components: a SBD file containing the file definition, a SBF file containing the data, and a variable number of index files. You will need to copy each of these in turn.

You may find it easier to use AmigaDOS or Workbench commands to back up your files. You could then, for example, copy all the files in a directory or all the component files of a single database file with just one command.

For example, if you have exited from Superbase, you could copy the Address database file to a floppy disk, using the AmigaDOS command:

copy Address.#? to df0:

The best way to backup database files from the hard disk is to use a tape streamer or special hard disk backup utility program. If these are not available, files should be copied from the directory where they are stored to a floppy disk. One of the purposes of backing up data is to insure against a hard disk failure; so it is not sensible merely to copy database files from one directory to another.

# Operating System Commands

The Utilities menu includes five commands which are the same as or similar to the disk or file management commands provided by AmigaDOS: Directory List, Type, Delete, Rename and Copy. The advantage of using these instead of the equivalent AmigaDOS commands is that you do not have to exit from Superbase.

## Directory List

This produces a listing of the current directory, in sorted order. The name of the directory and the date are also given.

## Type

 The purpose of Type is to allow you to view ASCII text files, such as Export files, SBD files, or Query files (SBQ files). You cannot use it to view program files, database files, or text files which contain control characters. This last category includes text files created in the Superbase Text Editor (SBT files).

When you select Type, Superbase displays a requester which lists the files for the current directory. You can then pick the file whose contents you want to view. As Superbase displays or prints the contents of the file it ignores tab characters. If the file contains non-printable characters (such as control characters), Superbase will display an error message and stop the display.

## Delete

This command deletes a file from disk. Note the difference between Delete and the Project menu command Remove. Remove deletes the current SBF (a database file) file together with its associated SBD file and index files. Delete deletes only one file at a time.

## Rename

This command renames a file on disk. When you have selected a file from the list, it displays another requester where you type in the new name.

## Copy

This command copies a file to another directory or another disk. After you have selected a file from the requester list, it displays another requester where you type in the name under which the file is to be copied. A file can be copied to the same directory under a new name or to a different directory under the same name or a new name.

# Documentation

At any time during a session with Superbase, you can use the Status commands on the Utilities menu or the Print command on the Project menu to obtain information about the system as a whole or about individual database files.

## Status File

This command provides a snapshot of the status and statistics of the current file.

Statistics include: current index, number of records, number of fields, number of indexes, block size and file size. Also detailed are the field names, attributes, format, and Page View co-ordinates for each field, and any calculation or validation formulas. Finally, you can see whether a password has

been set, the type of read/write privileges for the file, and the file's valid/invalid status.

• If the SET FILE MAX ON command has been used to allow Superbase to create and use extended files – those with more than 16 million records – the file size indicator is followed by the letter E.

## Status System

This provides a summary of the data described above for all open files, but excluding statistics for the individual files.

Additional data is shown for these items: Amount of free memory, amount of free disk space, current directory, number of files open and the number of files available.

## Project Print Command

The Print command on the Project menu allows you to output to the printer information that would appear on the screen if you used the Utilities menu. You can obtain printouts of:

**Directory List**
**Status File**
**Status System**
**Disk File**
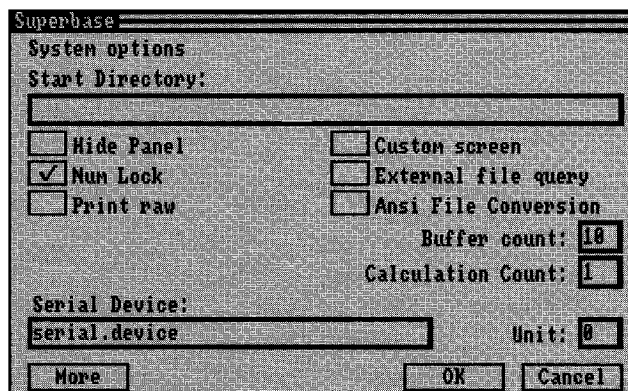
Check that the printer is switched on, loaded with paper and on-line, then choose the required Print command from the Project menu.

# 31 CUSTOMIZING YOUR SYSTEM
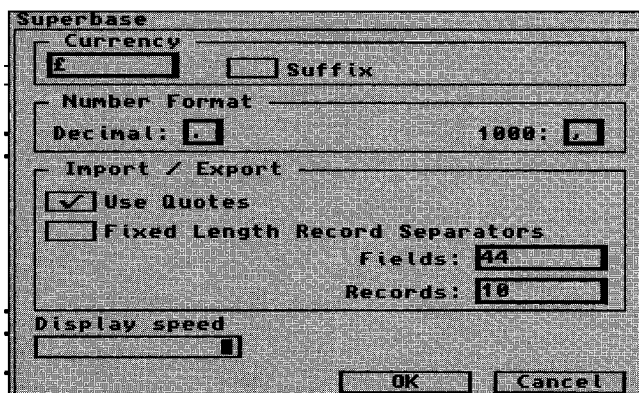
There are a number of Superbase features that can be changed to suit your individual preferences. Most of these are set using the System Options command on the Set menu. Selecting System Options presents you with this requester:

```
┌─Superbase────────────────────────────────────────┐
│ System options                                    │
│ Start Directory:                                  │
│ ┌───────────────────────────────────────────────┐ │
│ └───────────────────────────────────────────────┘ │
│   ┌─┐Hide Panel          ┌─┐Custom screen         │
│   │✓│Num Lock            └─┘External file query   │
│   └─┘Print raw           └─┘Ansi File Conversion  │
│                           Buffer count: [10 ]     │
│                        Calculation Count: [1 ]    │
│   Serial Device:                                  │
│   │serial.device                      │  Unit: [0]│
│   ┌──────┐              ┌──────┐┌────────┐         │
│   │ More │              │  OK  ││ Cancel │         │
│   └──────┘              └──────┘└────────┘         │
└───────────────────────────────────────────────────┘
```

If you click on the More button a second requester is displayed, allowing you to set further system options, as follows:

```
┌─Superbase────────────────────────────────────────┐
│  ┌─ Currency ──────────────────────────────────┐ │
│  │ │£        │    ┌─┐ Suffix                    │ │
│  └─────────────────────────────────────────────┘ │
│  ┌─ Number Format ─────────────────────────────┐ │
│  │ Decimal: │.│              1000: │,│          │ │
│  └─────────────────────────────────────────────┘ │
│  ┌─ Import / Export ───────────────────────────┐ │
│  │ │✓│ Use Quotes                              │ │
│  │ └─┘ Fixed Length Record Separators          │ │
│  │               Fields: │44     │             │ │
│  │              Records: │10     │             │ │
│  └─────────────────────────────────────────────┘ │
│  ┌─ Display speed ─────────────────────────────┐ │
│  │ │              ■│                            │ │
│  └─────────────────────────────────────────────┘ │
│                      ┌──────┐┌────────┐          │
│                      │  OK  ││ Cancel │          │
│                      └──────┘└────────┘          │
└───────────────────────────────────────────────────┘
```

The Import/Export separators are covered in Chapter 21 Import and the External File Query option is explained in Chapter 28 Integrating Text and Images. The remaining options are described under their own section headings further on in this chapter.

## The S:SUPERBASE.INI File

Whenever you click OK in the System Options requester, Superbase stores your preferences in the S:SUPERBASE.INI file. As well as storing various details of your configuration, this file contains a set of parameters which are used in various Superbase functions.

The next time you run Superbase, it will set the preferences you have specified in the S:SUPERBASE.INI file. In effect, this allows you to personalize Superbase so that it is always set up in a way best suited to your requirements.

The S:SUPERBASE.INI file can be viewed with the Type command on the Utilities menu, and contains the following parameters:

Workbench/custom screen
Use/bypass preferences
Buffer count
Calculation count
Decimal symbol
Thousands separator
Default number format
Default date format
Currency symbol
Position of currency symbol
Import/export field separators
Import/export record separators
Import/export quotation marks
Communications parameters
Text margins
Auto-dial number
Modem initialization
Preferences printer initialization
Default directory
Display/hide control panel
Num lock on/off
ANSI conversion on/off
Printer control file
Serial device name
Seerial device unit number
Form Designer ruler parameters

Some of the parameters in this list are not set in the System Options requester. The default number and date formats, for example, are stored in the S:SUPERBASE.INI file, but are set using their respective Set menu commands.

**Note**      Direct editing of the S:SUPERBASE.INI file is possible and does allow you to specify your own currency symbol of up to four characters. However, an invalid parameter file will cause Superbase to shut down on start up.

S:SUPERBASE.INI is a text file and can be edited in the Superbase Text Editor or in any wordprocessor. Here's an example of how the S:SUPERBASE.INI file stores some of the Superbase preferences:

```
Fieldsep=44
Recordsep=10
Quotes=3
Start=work:sbdata
Buffers=10
Calcs=1
Numeric=-999999.00
Date=dd-mmm-yyyy
Panel=1
```

**Note**   Any change to any item stored in S:SUPERBASE.INI that is saved results in the current settings of all such items automatically being saved.

Other Superbase preferences stored in the S:SUPERBASE.INI file include definitions of the text strings to be used for month names (long and short forms) and day names when dates are displayed or printed. For example:

```
Months=January,February,March,...
MMM=Jan,Feb,Mar,...
Days=Thursday,Friday,Saturday,...
```

If you wanted to display dates in a foreign language, you might, for example, substitute the lines:

```
Months=janvier,février,mars,...
Days=jeudi,vendredi,samedi,...
```

Note that if you redefine the day strings, you must start with Thursday.

# System Options

## Ansi File Conversion

This option only affects text files which contain characters with codes over 127. In most cases, these are accented characters or other instances of foreign character sets. If you do not use text files which contain characters whose codes extend beyond 127, you can ignore this option.

Superbase on the IBM PC uses the OEM/IBM character set rather than the Ansi set adopted by Workbench applications. Both sets have the same range of characters but use different values for characters with codes in the range 128 to 255.

Ansi File Conversion is relevant only if you exchange data files between computers of various makes. If your data files are used on Amiga computers only, you can avoid unnecessary file-processing activity by ensuring that the

Ansi File Conversion button in the Set System Options requester is not checked.

If you use Ansi File Conversion, its operation depends on the type of file being processed and on the process itself, as follows:

- Any text, program, export or import file, or file with a filename extension of SBD, SBV, SBQ, SBB or SBK that is output (saved) by Superbase is converted to OEM/IBM format.

- Any text, program, export or import file, or file with a filename extension of SBD, SBV, SBQ, SBB or SBK that is input (loaded) by Superbase is converted from OEM/IBM format.

- Database files (with filename extensions of DBF) are converted to OEM/IBM format when saved, and converted from OEM/IBM format when read, by Superbase version 2 and earlier.

- Database files (with filename extensions of DBF) are not converted by Superbase version 3 and later. Data files are saved in Ansi format and are assumed to be in Ansi format when read.

The Ansi character set is used when you handle text files in any of the following Superbase operations:

- Displaying text files on screen using the Utilities Type command.

- Importing and Exporting DIF files or ASCII delimited and fixed length files.

- Loading and saving non-Superbase files with the Text Editor.

## Currency Symbol

The Currency box allows you to specify the currency symbol to be used for numeric fields whose format includes a currency symbol. Click in the box and then type in the symbol or currency abbreviations (up to four characters) you require. The selection is global – you cannot have different currency symbols for different fields, although you can choose to print individual characters as such, e.g. '$'.

Normally, the currency symbol appears at the front of a number. Click in the Suffix check box, if you wish the currency to appear after a number.

## Number Format

Use the Number Format options to specify the characters that will be used to indicate decimal numbers and to separate thousands. In the Decimal box, you may enter a standard decimal point, a comma, or a space. For the thousands separator, you may enter either a comma or a space.

## Display Speed

This feature controls the amount of time that a record or picture is displayed on the screen before the next one is displayed. The feature is used in two situations:

*   When scrolling records in Table View or Record View, in either Fast Forward or Rewind direction.

*   When External Files are being displayed in Fast Forward or Rewind direction, or in a Query.

To alter the display speed simply move the pointer and click the vertical bar. You can then drag the bar to the left to make the display speed slower, or to the right to make it quicker.

The current setting of Display Speed is not stored in S:SUPERBASE.INI , so remains effective only until you exit from Superbase. The default speed setting is the fastest.

## Hide Panel

This option allows you to choose whether the browsing controls panel at the bottom of the window is displayed. Select the check box if you do not want the panel to be displayed.

You will then have to use the keyboard equivalents for browsing. See Chapter 33 Command Reference Summary.

## Calculation Count

Use this option to set the number of times calculations are performed. Enter the number in the box and press ENTER . See Chapter 8 Derived Values for more information.

## Buffer Count

This option varies the amount of memory Superbase uses to store record data. Increasing the number of buffers cuts down the number of times Superbase accesses the disk drive, and so increases the speed of the program. However, a very large number of buffers gives Superbase more work to do checking their contents in certain operations, and so slows the system down. Each buffer requires 512 bytes, so the total memory allocated to buffers can soon mount up. It is best to compromise: the recommended number to use is between 10 and 20.

## Start Directory

This option allows you to specify which directory will be the current directory when you start a session with Superbase. If you do not specify a start directory, you will start in the same directory as you were in when you loaded Superbase.

You will use this option if you store your data files in a different directory from the Superbase directory. If you specify the data directory as the start directory, the files it contains will be available as soon as Superbase has loaded.

To specify the start directory, enter the path name for the directory. For example:

    :sb4/sbdata

This line instructs Superbase to start up in the **sbdata** directory, which is a subdirectory of the **sb4** directory.

    df0:

This specifies that the start directory is on the disk in drive df0:.

# Date/Time Format

Date/Time Format allows the user to set the format for any dates and times that are not stored in fields. The system variables TODAY and NOW are displayed in the format set with this command, as are the the results of calculations involving date and time fields.

Date and time fields are displayed in the format that was set in the file definition. But, as with· Number Format, you can force Superbase to display these fields in the current format by enclosing the field name in parentheses.

This command is also used to set the system calendar and the system clock.

The Date Format requester and the procedure for setting a date or time format are described in Chapter 3 Defining New Files.

## Numeric Format

The Number Format command allows the user to set the format for any numbers that are not stored in numeric fields. This includes numeric variables, numeric constants, and the results of calculations involving numeric fields.

The data in a numeric field is normally displayed in the number format that was set in the file definition. But if the field name is enclosed in parentheses, it will be displayed in the current number format. For example, to display the field Annual Interest in the current format using a Query, you would enter:

    (Annual Interest)

in the Query Fields command line.

The Number Format requester and the procedure for setting a format are described in Chapter 3 Defining New Files.

# 32 LEARNING TO USE SUPERBASE

This chapter contains 11 lessons designed to give you practice in using Superbase. The lessons cover those database management activities that are performed most frequently, but do not explore the more advanced area of working with multiple files. This is discussed in Chapter 17 Multi-file Applications.

The lessons begin with setting up a new file and entering some data into it. Next we cover the browsing controls and the use of the Filter to select records. After showing you how to import a file of example data we explain how to select fields and switch indexes, then print out a list. We then show you how to print mailing labels and do a mail merge and an update. A guided tour of Superbase's query feature concludes the chapter.

**Prerequisites.** Before attempting the tutorials you should familiarize yourself with certain aspects of the system. The first chapter in the manual, Getting to Know Superbase, explains some important features of both Superbase and Workbench, so please read it before proceeding.

You may also benefit from exploring the example application included with the package, and we advise you to copy the example data into the same directory as you use when going through the tutorial.

You should copy the example file **Accounts.csv** into your tutorial directory before starting Lesson 6.

# Lesson 1: Defining a File

In this example you are going to set up a file to hold details of bank accounts. The file includes examples of several types of field.

Briefly, there are five stages to the tutorial:

1. Select New File from the Project menu and type in the file name.

2. Type in a password.

3. Enter the field names and details.

4. Specify index fields.

5. Save the file definition.

6. Print out the file definition.

Start by selecting New File from the Project menu.

## New File Name Requester

Superbase displays the following requester:



You now have to type in a file name. For this tutorial we are going to create a file called **Accounts**.

Type in 'Accounts' and click on OK. Superbase now displays the Password requester. Type in a single password, so that you will have all levels of privilege. Keep it simple and easy to remember as you'll need to enter the password each time you open the file. Note that passwords are case sensitive.

Once you've typed in your password click OK. Superbase displays the File Definition requester:
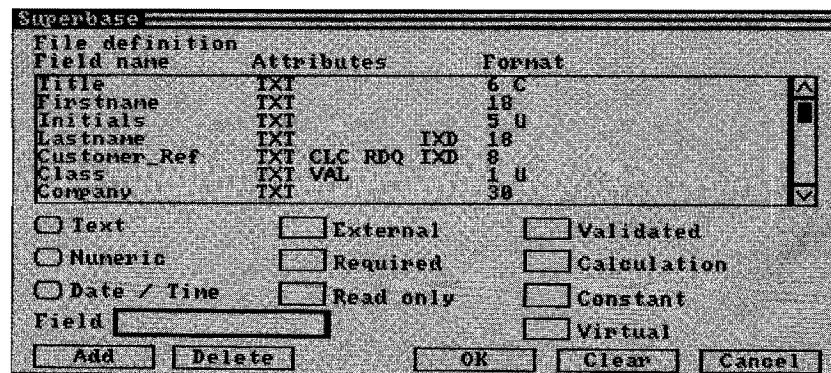
The File Definition requester shows all the controls you will use to complete the process of defining the fields for the file. For each field, you just type in the name, set the desired characteristics, and add it to the definition so that it appears in the large panel at the top of the requester. When you've finished adding fields, you specify which ones are to be indexed. Then you can start adding data to the file.

The fields for the Accounts file are:

* Reference
* Name
* Number
* Type
* Bank_name
* Bank_code
* Date_opened
* Balance
* Interest Rate
* Interest

## Reference

This is going to be the reference number for each record and will be a required field. We'll set up this field as a text field because it will contain both alphabetic and numeric characters.

The procedure for defining a field is quite simple. First the field name must be typed into the field box. If the insertion point is not already showing, click in the Field box to make it appear, then type in the field name 'Reference'.

Next, you have to define this field as a text field, so click the Text radio button. Superbase then displays the Text format requester.
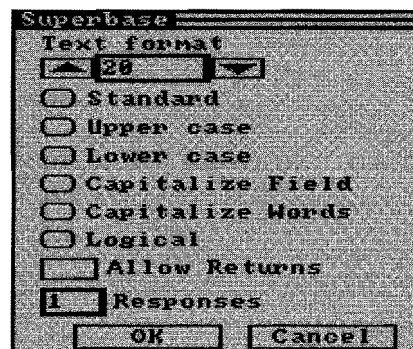
This requester enables you to set the length of each Text field and various other attributes such as the number of responses the field will accept. At the top, you will see the field length box. The default value is 20 characters, which is the value you can see on your screen. The **Reference** field only needs a length of 5 characters, so click in the editing box at the top of the requester, erase the existing figure using first the RIGHT key then the Backspace key, then type in '5' and complete the entry by pressing ENTER .

To ensure that any letters typed into the field are always converted to uppercase, click the Upper case radio button.

We can ignore the other Text format options, so you can now click the OK button to clear the screen and return you to the File Definition requester.

Now we have to make the field a Required field, so that it can't be left blank by mistake. So click the Required check box. Superbase places a tick in the box to show that this attribute has been selected.

This field is added to the file definition by clicking on the Add button at the bottom left of the requester.

Do this now and watch the screen.

Superbase displays the field name in the File Definition Panel along with its attributes and field length. Check that these details are correct before moving on to the next field.

## Name

The next field to enter is **Name**, which will also be a text field, but not a required field. This field will hold the name of the account.

Click in the field box if the insertion point is not already there, then type in **Name**. Click the Text radio button to call up the Text format requester. You'll see that Superbase again displays the field length of 5 characters, which is the length we used for the previous field, but we want a field length of 20 characters. Use the RIGHT and Backspace keys to delete the '5', type '20' and press RETURN .

We do not need the upper case option for this field, so ensure that the Standard radio button is selected. When you've done this, click OK to exit back to the File Definition requester. Now you can add this new field by clicking on the Add button. If you then look at the File Definition Panel you will see that the field you have entered is shown on the second line.

## Number, Type, Bank_name, Bank_code

You can create these four text fields just as easily as the last two. For each field, follow this routine:

1.  Enter the field name,

2. Click the Text radio button.

3. Enter the Field length:

|  |  |
|---|---|
| Number | 10 |
| Type | 1 |
| Bank_name | 20 |
| Bank_code | 10 |

4. If the Standard radio button is not selected, select it.

5. Click OK.

6. Click Add.

## Date_opened

This field is different from those you've created so far: it's a Date type field.

Start off the same way, by entering its name into the Field box.

Now you must define the format in which the dates typed into the field are to be displayed. Click the Date / Time radio button. Superbase displays the following requester:
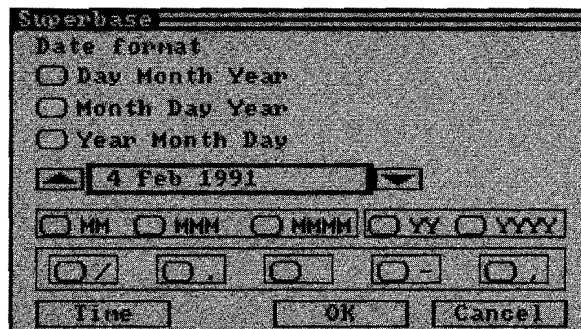


If you look at the requester you will see the current system date is displayed in the default format. This will vary depending on the format set in your S:SUPERBASE.INI file. The display of the system date is used here as a guide to the way the field will display dates. You can change both the day/month/year order and the format of the date. Every selection is reflected at once in the date display, so you can experiment freely until you are satisfied with the results.

Select the order 'Month Day Year' (this is necessary even if your national conventions are different for the rest of the tutorial to work properly). Then choose the style of month and year display you prefer. Choose a separator from the next group of options.

When you've finished making selections, exit from the Date format requester by clicking OK. Finally, add the field to the file definition by clicking on the Add button.

## Balance

This field will contain the balance for each record within the **Accounts** file and therefore we must define it as Numeric. Once again the field name is entered in the same way as before, by typing the word Balance into the Field box. However, this time you must click the Numeric radio button. Superbase displays another requester which looks like this:
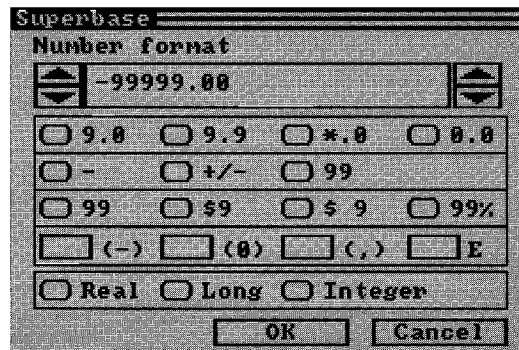


As with the Date format requester, Superbase will show the default format. We will change this to include a currency symbol and a separator for thousands.

Move the pointer and click the $9 radio button, then click the (,) check box.

Once you've done that click the OK button, and when you're back at the File Definition requester click the Add button to add this field to the file definition.

## Interest_rate

This field will also be set up as a Numeric field, and we will introduce a validation check to limit the range of values that can be entered. We'll define this range as being greater than zero and less than 15.

Type in the field name in the usual way, then click the Numeric radio button. Notice that Superbase has carried over the format and the features from the previous numeric field. This can be very useful when you have to create many fields with the same format one after another.

For **Interest Rate**, we will need to change the Number format. First, the number is not going to need provision for a sign, as it's never going to be less than zero. So, click the right-hand sign button – the one that does not show a sign symbol. This field uses a percent format, so click the 99% symbol button. Then deselect the (,) check box by clicking on it.

Next you need to shorten the number format. Click the lower of the two arrow buttons on the left- hand side of the number format panel. As you do this you will notice that the number format changes from 99999.00% to 9999.00%. Click the button until the number is reduced to 99.00%.

Once you've done that click OK. Now click the Add button. Superbase adds the **Interest_rate** field to the File Definition Panel, and clears the Field box.

Next you're going to add a range check to this field. First you must make the **Interest_rate** field the selected field. Click its name in the File Definition Panel to reselect it and copy its name into the Field box. Now click the Validated check box: Superbase displays the Validation Formula requester.
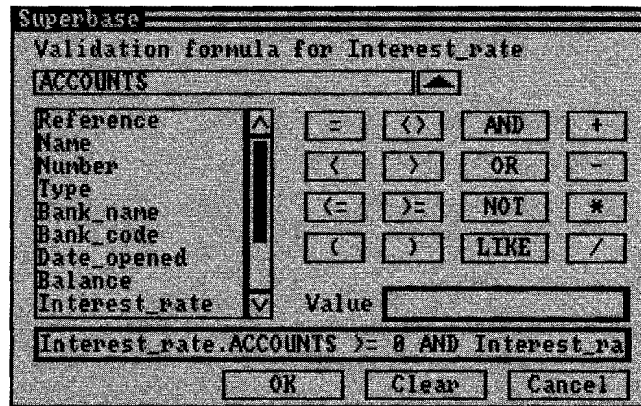


This requester enables you to create a validation formula that will be used by Superbase each time a value is typed into the **Interest_rate** field. The formula will ensure that every interest rate is within the permitted range.

The formula you want is:

Interest_rate.Accounts >= 0 AND Interest_rate.Accounts <= 15

Do this in stages. Move the pointer into the Field list box and click **Interest_rate**. Next, click the >= button. Now you have to enter 0 in the Value box. The insertion point is automatically positioned in the Value box, so type in '0' (omit quotation marks) and press ENTER to copy the entered value into the large text box.

Your Validation Formula should read like this:

Interest_rate.Accounts > = 0

To make a link to the next part of the Formula, you click the AND button. Now repeat what you did before: click **Interest_rate,** then (slightly different) on the <= button, and then in the Value box, where you type '15' and press ENTER . Your formula should now look like this:

Interest_rate.Accounts > = 0 AND Interest_rate.Accounts < = 15

If you've made any errors, you have a choice of how to correct them. Either click directly in the text box and edit the formula directly, or click Clear and repeat the steps above.

Once the formula is correct, click the OK button, and Superbase will return to the File Definition requester. Notice that Validation is now shown as selected.

Superbase updates the File Definition Panel to show VAL under the Attributes column, but it does not show the Validation formula; you can use Utilities Status File to look at it later.

## Interest

This is the last field we need to enter for this file, and it's going to be a calculated field. This means that Superbase will calculate a result for the field based on the values of the fields referred to in the formula, in this case **Balance** and **Interest_rate**.

So, click Clear to remove **Interest_rate** from the Field Box, and then click in the box and type in the field name **Interest**. Click the Numeric radio button. The format needs to be reset from the format used for **Interest_rate**. Click the $9 and (,) buttons to set the currency sign and thousands separator, and the - button to provide for a minus sign. Then click the upper of the two arrow buttons to the left of the number format panel three times to expand the format, which should look like this: -$,99999.00.

When it's right, click OK to exit back to the File Definition requester. Click Add to add the field to the file definition, then click on the field name **Interest** in the File Definition Panel to reselect it.

Next, click Calculation. Superbase displays the Calculation Formula requester, which works exactly like the Validation Formula requester you have just used.

What we want Superbase to do is to calculate how much interest is due for each balance based on the current **Interest_rate**, and then display this value in the **Interest** field.

So, the formula to calculate the amount of interest is:

Balance.Accounts * Interest_rate.Accounts / 100

You should be able to do this without detailed guidance. Click Clear to empty the Text box. Now click the field name **Balance**, the * button, the field name **Interest_rate**, and the / button. Then click in the Value box and type in the value '100'. Press ENTER. The Formula should read as above. Once it's correct you can click the OK button.

That completes the entry of the fields, so now you can click OK in the File Definition requester.

## New Indexes

Superbase now clears the requester from the screen and displays the New
Index requester.
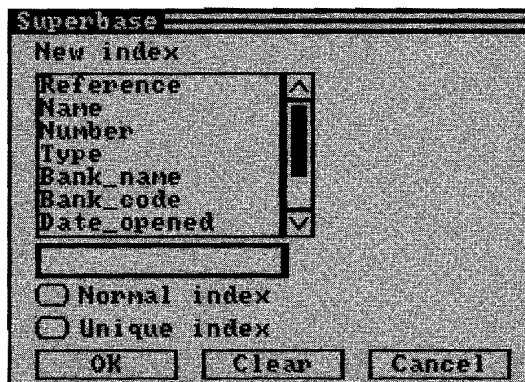


This is the last part of the process of setting up a file, which is to specify the
field names you want to use as indexes. If you look at the screen you will see a
list of the field names you have just entered. What we have to do now is to
decide which fields to use as indexes. For the **Accounts** file we want indexes on
**Reference, Bank_name, Date_opened**, and **Balance**.

You can only specify one index at a time. So, move the pointer and click
**Reference**. As you do this Superbase displays this field name in the box at the
bottom of the list. Also notice that Superbase has highlighted the Normal
radio button, indicating that duplicate index entries may be allowed. Since the
purpose of the **Reference** field is to provide a unique identifier for each record
in the **Accounts** file, the index should not allow duplicate entries. Click the
Unique radio button to indicate this. Now click the OK button. Superbase
redisplays the list of fields without the reference field. You can go on and
specify as many more fields as you want as indexes.

Repeat the exercise for **Bank_name, Date_opened**, and **Balance**, this time
creating a Normal index for each one. Notice that each time Superbase
redisplays the New Index requester, the field name you used to specify the
previous index has been removed.

When you have finished selecting fields for indexing just click the OK button.
This informs Superbase you have finished making your selections and it can
proceed to save the File Definition and create the Index Files.

Once the indexes have been built Superbase displays the file in the format of
the current view. The file will be indexed on the first index you specified, in
this case **Reference**, as shown by the message on the title bar at the top of the
window. You can toggle to the other indexes by pressing the '–' key.

Superbase also displays a blank record showing the field names you set up in the File Definition.

# Lesson 2: Entering Data

In this example you will use the **Accounts** file you created earlier to create, edit, duplicate and remove records.

Select Record View from the Set menu for this exercise.

## Opening a File

If the **Accounts** file that you created in the last lesson is not already open, you need to open it. Select the Open File command from the Project menu. Click **Accounts** in the left-hand list box, then click OK.

When you have opened the file, you will see the field names down the left-hand side of the work-area, with no data on the right-hand side. This is because the file is empty.

## Adding New Records

Press the right mouse button and select the Record menu. This is the menu you use when you are creating, duplicating or saving records. The command to use when you are entering a record for the first time is New, which is the first command on the menu. Select New now.

If you look at the screen, you will see that Superbase is displaying the cursor in the first field of the record. The end of the field is shown by a vertical bar.

You are now going to enter your first record into a Superbase file. Here is an example record:

|  |  |
|---:|---|
| Reference: | R001 |
| Name: | John B. Sloop |
| Number: | 0405332 |
| Type: | D |
| Bank_name: | National City |
| Bank_code: | 401362 |
| Date_opened: | Jul 6, 1986 |
| Balance: | $2400 |
| Interest Rate: | 5.67 |

Start by entering the **Reference**, R001. When you've typed the example data, press ENTER . The cursor moves to the next field. You can then continue and type in the rest of the data. Remember to make sure you are entering the data against the correct field. Don't worry about making mistakes, as these can be corrected as you go along by using the DEL or BACKSPACE keys.

When you press ENTER after typing the **Interest_rate**, Superbase calculates and displays the **Interest** automatically. Because there are no more fields requiring data, Superbase now displays a requester asking whether you wish to save the record. You may click on OK, but most people just press ENTER at

this point because their hands are still on the keyboard. Do this now. After saving the record you have just entered, Superbase presents a requester giving you a choice between entering another new record or ending data entry.

Type in another two records:

| | |
|---|---|
| Reference: | R002 |
| Name: | Sarah Armstrong |
| Number: | 6902237 |
| Type: | D |
| Bank_name: | People's Co-Op |
| Bank_code: | 336512 |
| Date_opened: | May 12, 1984 |
| Balance: | $5000 |
| Interest Rate: | 6.75 |

| | |
|---|---|
| Reference: | R003 |
| Name: | Mary Garcia |
| Number: | 9784421 |
| Type: | C |
| Bank_name: | National City |
| Bank_code: | 401362 |
| Date_opened: | Feb 21, 1989 |
| Balance: | $749 |
| Interest Rate: | 5.67 |

You can enter these records in exactly the same way as you did before, but remember to look at each field as you type in the data, to make sure it's the right one. Remember also to save each record before moving on to the next.

When you've entered all three records, end data entry.

# Lesson 3: Editing a Record

Suppose you want to change some of the details in John B. Sloop's record. How do you do it?

First of all you have to select his record. Press the left direction key, to display the previous record in the file, as many times as necessary to display the required record.

Select Current from the Edit menu. Superbase positions the insertion point in the first field of the record.

Look at the record, and find the **Date_opened** field. You are going to change this from 'Jul 6, 1986' to 'Jul 7, 1986'. Press ENTER six times to move the insertion point down to the **Date_opened** field. Use the right direction key to move along until the insertion point is immediately to the right of '6' and press Backspace to delete it. Then type '7' and press ENTER . That's it, you've now edited the record. Save it by selecting Save from the Record menu.

The example you've just worked through is an easy standard procedure for editing a record. It is often easier to edit a record simply by clicking in the field to be changed, making the necessary changes, then saving the record with the AMIGA+S shortcut. Records may be edited in Record and Page Views, but not in Table View.

There are many more editing features available to you than are described here. See Chapter 6, Entering Data, for more details.

## Duplicating a Record

If you want to create a new record that has very similar data to another, you can do so using a convenient shortcut, the Duplicate command. What you're going to do now is duplicate Sarah Armstrong's record and create one for her sister. So select Sarah's record by pressing the right direction key.

Now move the pointer to select the Duplicate command from the Record menu. Superbase displays a message to say that a copy of the record has been made in memory. You can now edit it.

You need to change the information in five of the fields, **Reference, Name, Number, Date_opened**, and **Balance**.

| | |
|---|---|
| Reference: | R004 |
| Name: | Janice Armstrong |
| Number: | 6902551 |
| Date_opened: | Mar 15, 1984 |
| Balance: | $3200 |

If you look at the screen you can see the cursor is already at the **Reference** field. Use the right direction key to move to the right of the data, press BACKSPACE to delete the '2' and type '4'. When you've done that press ENTER

to move the cursor to the next field, **Name**. At the beginning of the field, press the DEL key five times, then type 'Janice'. Now change the **Number**, **Date_opened**, and **Balance** fields on your own.

That completes the editing for this record so you can now save it in the usual way.

# Lesson 4: Using the Browsing Controls

The browsing controls at the bottom of the Superbase window give you immediate access to your data. These controls, modeled on those of a VCR or tape recorder, are designed to be used with the mouse, but some have keyboard alternatives. We mention each alternative the first time the browsing control is described, but subsequently assume that the mouse is being used.

## Select First Record

When you want to display the first record in the file click the First Record button, third from the left.

Do this now, and you will see record R001 appear.

The current index determines the order in which records appear. Select the Project Open Index command, and click the **Date_opened** fieldname in the left-hand list. Click OK. Superbase shows in the title bar that the **Accounts** file is now indexed on **Date_opened**. However, the current record is the same. Click the First Record button again, and record R004 will appear. This is the record with the earliest date in the **Date_opened** field.

## Select Next Record

This very frequent browsing action is performed by clicking the right arrow button, seventh from the left, or pressing the RIGHT direction key.

Each time you click Next Record, Superbase displays the next record in the order of the current index, shown in the window title bar.

Do this now, and record R002 appears. Repeat the action twice to show records R001 and R003 in that order.

Click Next Record again. The 'End of File' message appears temporarily, informing you that there are no more records to be displayed.

## Select Previous Record

The opposite to the last command, Previous Record displays the record before the current record in the order of the current index.

Click Previous Record, fifth from the left, or press the LEFT direction key. Superbase displays record R001. Repeat the action three times to show records R002 and R004, followed by the 'End of File' message.

## Select Last Record

You can choose the last record in the file by clicking the Last Record button, ninth from the left.

Do this now, and Superbase displays record R003.

### Rewind: Scanning in Descending Order

The VCR model used for the browsing controls includes an equivalent to the Rewind function. By clicking on the Rewind button, fourth from the left, you instruct Superbase to display the records one after the other, starting from the current record and moving in the reverse order of the current index.

With record R003 on the screen, click Rewind. Superbase displays records R001, R002, and R004 followed by the 'End of File' message.

### Fast Forward: Scanning in Ascending Order

The opposite of Rewind is Fast Forward. This command allows you to scan through the file in the order of the current index.

Click the Fast Forward button, eighth from the left. Superbase displays the remaining records and then the 'End of File' message.

### Retrieving Individual Records: Key Lookup

We use the term 'key lookup' to describe the action of retrieving a specific record from a database file by means of an index key.

Key lookup works by asking for a value for an indexed field, which Superbase then uses to read the record from the disk. The process resembles using a book index to find the page number for a particular reference, then turning to the correct page. Key lookup is a very fast way to retrieve individual records, but of course you can only enter a single field value (unless you have set up indexes based on multiple fields – see Chapter 4 Creating Indexes).

Press the '–' key to cycle through the available indexes until the title bar shows that the Accounts file is indexed on Reference. The key lookup function always uses the current index.

Click the Key Lookup button, third from the right, or press the '?' question mark key. Superbase displays a requester asking you to enter the Key Value. Type 'R002' (omit quotes), and press ENTER. Superbase displays record R002.

Repeat the action, this time typing 'R004'. Again, the matching record is displayed.

Now, repeat the Key Lookup button, but just type in 'R' and press ENTER. Superbase does two things:

- Informs you that there is no key with the exact value 'R';
- Displays the record closest in sequence after 'R' which in this case is record R001.

This very useful feature of partial key entry allows you to move very rapidly to any section of a file, and then use the other browsing controls to look at nearby records.

Now press the '–' key again to change to the Date_opened index. Click the Key Lookup button, type '2/21/89' and press ENTER . Superbase displays record R003. (Note that you can type in a date in any format provided the day/month/year order corresponds to that of the default date, which you can view with the Set Date Format command.)

## Pause and Stop

The two leftmost buttons in the browsing controls allow you to interrupt the display temporarily or permanently. Their respective keyboard equivalents are SPACEBAR and CTRL+C . These controls are mainly used with Fast Forward and Rewind, and are useful when you are scanning through large files.

- Note that you can also vary the speed of the display from the Set System Options requester.

## Current Record

The central Current Record button is mainly used when the screen is temporarily displaying non-record data. Examples are the output from the Utilities Status File or Utilities Type commands. Current Record re-reads and displays the current record.

For keyboard equivalents you may use either the UP or the DOWN direction key.

# Lesson 5: Working with Filters

The Superbase filter system allows you to choose subsets of records from your file. Filters are used throughout Superbase, so it is important that you explore and master filter techniques. Before continuing, press the '–' minus key to switch to the Reference index, which we shall be using later in the lesson.

## How Filters Work

As a first step, click the Filter browsing control button, second from the right, or press the = equals sign key.

Superbase displays a requester showing the fields from the current file in a list box, together with a set of operator buttons. There is a text box extending across the requester, in which your selections will be combined to create a filter command line.

When you click OK, Superbase displays the first record in the file that matches the criteria you have defined in the filter command line. You may then use the other browsing controls to review the other records that match the criteria.

Most of the Process menu commands use the filter command line that is set up in the browsing filter as the default for their specific activities. This makes it easy, for example, to preview a set of records to be discarded using the browsing filter, then actually remove them with the same filter line as part of the Process Remove command.

Filter command lines are constructed from expressions that combine fieldnames, operators, values, and functions. An example of a simple expression would be:

Balance.Accounts > $500

More complex filters can combine several expressions with the AND, OR, and NOT keywords, and may draw on the full range of Superbase's DML programming language functions.

## Defining a Filter

We are going to search for those account records where the record balance is between $2000 and $4000. So we'll use the Balance field, the '>' and '<' operators, and the Value box that appears below the operators. You'll see the results of your selections appear as a filter command line in the text box.

Move the pointer into the list of fieldnames and click the Balance field. The fieldname and its file extension are copied into the text box: **Balance.Accounts**. Next, move the pointer to the right and click the '>' button. This also appears in the text box. Now you must type in '2000'. This goes in the Value box. Superbase has already activated the Value box by placing an insertion point in it, and while this is present you can press ENTER

without exiting from the Filter requester. Type in '2000' (no quotation marks).
Press ENTER – this is essential, or the contents of the Value box will not be
copied into the text box. Your filter command line should read:

Balance.Accounts > 2000

If it does not, click Clear and start again.

Next, you need the AND operator. Click the AND button and Superbase
copies it into the Text Box. Now you must click the **Balance** fieldname again.
This is important. Every expression must have 'two sides,' so that Superbase
can resolve it into either true or false, so a filter command line like this is not
allowed:

Balance.Accounts > 2000 AND < 4000

So, click **Balance**, as before, then follow it with '<', then type '4000' into the
Value box. Press ENTER . Your filter command line should look like the one
shown in the diagram below.
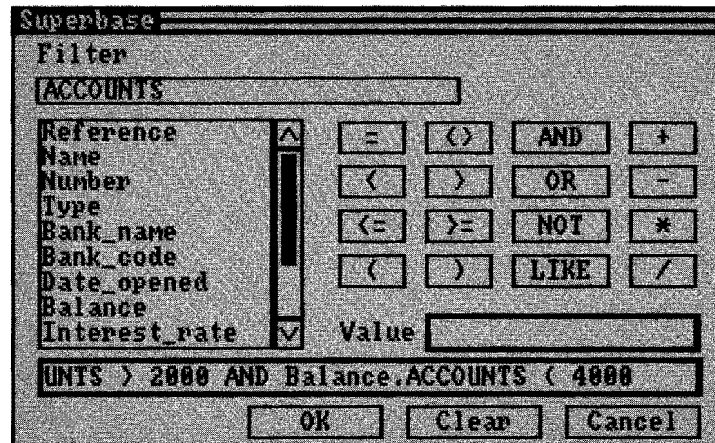


If there are any mistakes click Clear and repeat the steps above until you get it
right.

You have built your filter command line, so now you can put it into action.
Click the OK button. Superbase activates your filter and displays the first
record in the file that matches its criteria, which happens to be record R001.
Click Fast Forward. Superbase shows the other record that matches the filter,
R004.

Now de-activate the filter. Click the Filter button so that it's not highlighted.
Click First Record followed by Fast Forward. You'll see all the records in the
file appear as normal.

## Extending a Filter

Let's go back and add another condition. Click the Filter button to recall the requester. The filter command line is retained for you to modify.

Click the AND operator, then on **Date_opened**, then on '>'. Type '1/1/86' and press ENTER . Your line should read:

    Balance.Accounts > $2000 AND Balance.Accounts < $4000
    AND Date_opened > "1/1/86"

Click OK. This time Superbase finds only one record that matches all three criteria, record R001. If you try to select the next record you will see the 'End of File' message, which tells you that there are no more records that match the filter.

## Overriding the Filter with Key Lookup

The only browsing control that does not use the filter is the Key Lookup button. See this for yourself now. Select Key Lookup, then type in 'R003' and press ENTER . Superbase retrieves this record even though it does not match the filter criteria. However, the filter remains active and the other browsing controls will find only matching records.

Before going on to the next lesson, turn off the filter by clicking the Filter button.

# Lesson 6: Importing Data

The ability to import data into Superbase from another system allows users both to switch to the Superbase system without re-entering data, and to work effectively with other applications software. In this lesson we see how you set about importing a straightforward text file, which will allow us to expand the example **Accounts** file and make the following lessons more useful.

Before you continue, ensure that the supplied example file **Accounts.csv** is in the same directory as the **Accounts** file you created in the first lesson. If you have been using the Set Field Selection command, ensure that it is deselected (there should be no check mark against it when you display the Set menu).

Now, select the Process Import menu command. Superbase shows the Import File Type requester. Of the many possible selections, we want the default, ASCII Delimited Merge. The radio button for this is already highlighted, so you can immediately click OK.

Superbase shows a Filter requester like the one you used in the previous lesson. If you have carried straight on from the previous lesson, the last filter command line will be shown. Since you want to import all the records from the **Accounts.csv** file, you must remove it. Do this by pressing DEL if the line is highlighted, or clicking Clear if it is not. When the text box is empty, click OK.

**Note**    Throughout Superbase, whenever you are presented with a filter requester and you want to process all the records, click OK without specifying a filter command line.

Superbase shows a file selection requester. Change the pattern in the text box to show **\*.csv**, and press ENTER . The list of files should show **Accounts.csv.** Click this filename, then click OK.

It should take Superbase only a few seconds to read the data from **Accounts.csv** and create new records in the **Accounts** file. All the Accounts file indexes will be automatically updated.

Use the browsing controls to display records from the expanded file.

The Utilities Status File command shows how many records are in the new file.

# Lesson 7: Selection and Output

Superbase allows the user a very wide choice of selection options in addition
to the filter method of record selection that was covered in an earlier lesson.
You can select and arrange fields to suit your requirements, and then output
them in any order. In this lesson we shall see how you choose a group of fields
for output, and how this feature can be combined with index selection to
create an instant report.

This lesson includes a printing exercise. If you do not have a printer installed
for Workbench, or your printer does not work properly, you should try to
resolve the difficulties before starting the lesson, as we do not want you to
become distracted during it.

## Selecting Fields for Output

Choose the Field Selection command from the Set menu. Superbase displays a
requester with two list boxes. The left-hand box shows a list of the fields in the
**Accounts** file; the right-hand box, which shows the current list of selected
fields, is empty.

You are going to select a list of fields from the left-hand box. First, click on the
**Date_opened** field. Superbase copies the fieldname into the right-hand box.
Next click on the **Name** field. Again, Superbase copies the fieldname into the
right-hand box. Now click on these fields in succession: **Type, Balance,
Reference.**

Your list of fields should be:

      Date_opened
      Name
      Type
      Balance
      Reference

If your list differs, click Clear and start again. The Delete button removes the
last item from the list of selected fields.

Notice that we have altered the order of the fields. You may select fields in
any order with this feature.

When you have chosen the fields correctly, click OK. Superbase redisplays the
current record showing only the selected fields.

You have already practised switching from one index to another using both the
File Index Open command and the '–' minus key. Select the **Reference** index
and use the browsing controls to review the file. You will see that Superbase
now only displays data from the selected fields.

**Note** You can save the selected fields list in the file definition with the Project Save command. Next time you open the file, the Field Selection requester will show the saved list as the default.

## Showing Selected Fields in Table View

Table View is especially useful in conjunction with the selected fields list. For example, you can select just those fields you want to see in a sequence that suits a particular index order, then review them in Table View's columnar format.

Select the Set Table View menu command now. Superbase shows the current record across the top of the screen with the names of the selected fields above them.

Press '–' to cycle through the indexes until the **Date_opened** index is selected.

Select First Record (HOME).

Redisplay by selecting Current Record (UP direction key).

Select Fast Forward (PGDN).

Use other browsing controls to step through the file in different ways using Table View.

**Note** Table View is a form of output. Superbase does not retain all the data you can see on screen in memory, so you cannot point at a record in Table View and click to edit it. Editing is only possible in Record or Page View.

## Printing Selected Fields

The Project Print menu command allows you to print out data from selected fields and records in any of the three Superbase views: Page, Record, and Table.

First you must define your printer and page layout. Since users have a variety of printers and stationery, we shall not attempt to give detailed instructions here. The following procedure is an outline only:

1. Select Printer Setup from the Set menu, and if more than one printer is available, click the one you wish to use.

2. Click OK to select the indicated printer, and to proceed to the Page Setup requester.

3. Define your page size and the margins you want, in inch or centimeter values.

4. Click on the Form Feed button so that it is checked if your printer can make correct use of the form feed character (ASCII 12).

5. Click on the Single Sheet button if you want the printer to wait at the end of each printed page for you to perform some manual operation.

6. Select the font you wish to print in. Click on the upward arrow to the right of the font name box to cycle through the list of available fonts, and stop when the one you want is displayed.

7. You could change the Initialize sequence for your printer now, but you should not normally need to do so.

8. When you are satisfied with all the settings on the Page Setup requester, click OK.

You are now ready to print your selections. Choose the Project Print menu command. Superbase displays a Filter requester. Unless you want to select a group of records from the file, just click OK.

The next requester allows you to specify any index for ordering the output. The current index is the default, in this case **Date_opened**. Let's make use of the Descending order feature, which will cause the records to be printed starting with the most recent date. Click the Descending order check box.

When you're ready, and sure that the printer is on-line, click OK. Provided your printer has been correctly installed, Superbase will produce printed output.

# Lesson 8: Printing Labels

Label printing is an integral part of the Superbase system. You can define label formats to match practically all types of label stationery. Label formats are re-usable, and are stored on disk as files with an SBB extension.

In this lesson we shall use the **Accounts** file again. Although it does not contain address information the exercise is still realistic: labels are often used to identify folders for client records.

Ensure that you have the **Accounts** file open. The printer should also be connected and on-line, as for the previous lesson, but you can print on ordinary paper rather than label stationery if you prefer.

To start the exercise, select the Labels Edit command from the Process menu. Superbase now shows the Label Definition requester. The label definition procedure is very simple; in outline:

1.  Position the required fields on the label layout

2.  Define label dimensions

3.  Define print run parameters (number of copies, etc.)

4.  Test alignment

5.  Print labels

## Defining the Label Layout

If a selected fields list is in use it is shown as the default for the label layout. Since we will not be using this, click Clear to remove it.

Move the pointer into the field list box and click the **Reference** field. Superbase copies it into the label layout and moves the highlighted selection to the next position across. Click the **Name** field. (If you position a field wrongly, highlight it and click the Delete button on the requester.)

Click in the label layout panel to activate it. Now use the DOWN and LEFT keys to move the highlight to the position below **Reference**. In the fields list, click the **Date_opened** field. Then click in the label layout panel and use the DOWN and LEFT keys again, this time leaving a blank line below **Date_opened**.

Click the **Number** field, then the **Type** field, then move DOWN and LEFT and click the **Bank_name** field.

That completes the layout.

**Note**     Superbase appends the second, third and fourth fields on a line to the first field, separating fields with a space.

## Defining the Label Dimensions

The next task is to define dimensions. As we do not know the details of your labels, these instructions only provide an outline of the procedure.

First, set the number of characters for the left margin. This tells Superbase where to start printing the first column of labels. Do this by typing the required value into the box at the top left-hand corner of the label layout.

**Note**    All measurements are in characters, where a character has a fixed width defined by the font selected in Page Setup.

Next, define the number of characters to be printed on each label line. Type the value into the box above the label layout.

Now you must specify the gap between labels. This tells Superbase where to start printing the second and subsequent labels across. Type the value into the upper right-hand box.

Lastly, you must tell Superbase how many lines to print before starting the next row of labels. Type this value into the box at the lower left-hand side of the label layout.

## Defining Print Parameters

The label print parameters are four in number:

   Number of copies
   Number of columns
   Number of rows per page
   Number of lines for the top margin

These values are set in the lower left-hand area of the requester. First, define the number of label columns you want. There is no real limit, as no currently supported printer is likely to allow more than the 99 columns Superbase allows. Unless you have mounted multiple column label stationery on your printer, leave the number of columns set to one.

You may also define up to 99 copies per label. For this exercise, leave the value at its default of one.

The remaining two parameters are relevant only if you are printing onto separate sheets of labels – not if you are using continuous stationery. The number of rows per page tells Superbase to eject a page after that number of label rows have been printed. The top margin specifies the distance from the top of the label stationery to the start of the first label.

## Alignment Test

When you have defined all aspects of the label, it's advisable to test the alignment of your stationery in the printer, as even a slight error can result in pages of useless output. Position the stationery so that the print head will

begin printing on the first line of the first label (this may apply only to continuous stationery).

Now click the Test button on the requester. Superbase outputs one page of labels as you have defined them, using the character 'X' to show where data will appear on each label line.

You may need to adjust your label definition and retest before proceeding to the full print run.

## Printing

When you are satisfied with the label test, click OK on the requester. Superbase presents a Filter requester. Click OK unless you wish to select a subset of records.

Superbase begins to print data from the file onto the labels.

When label printing ends, Superbase returns to the work area, using the set of fields used for the label format as the selected fields list.

If your print run fails, for example because the label stationery jams, you do not need to repeat the whole print run. You can use the filter to specify an index key value at which to restart printing. If the current print run stopped after record R003, you would enter a Filter command line like this:

Reference.Accounts > "R003"

**Note**    To optimize performance, you should ensure that the specified field is the current index.

## Saving the Labels Format

When your labels print run has finished, return briefly to the Label Definition requester to save the format for reuse.

Check that all aspects of the label format are defined as you want them. Then click the Save button on the requester. Superbase requests a name for the labels format file. Type in 'Accounts' and press ENTER .

Superbase creates a disk file called **Accounts.sbb** which contains all the details of the format, but not the filter command line.

When you want to use an existing label format, select Labels Open from the Process menu and select from a list of SBB files.

# Lesson 9: Mail Merge

Instead of the usual address merge, we shall use the information in the Accounts file to produce a short memo for inclusion in a mailing to existing clients. Ensure that the Accounts file is the current file before continuing.

## Typing the Merge Document

First, you must open the Superbase Text Editor window. Select Text Editor from the Utilities menu, then click the sizing gadget near the top right-hand corner of the window to expand it to full size.

Type in the following text, pressing ENTER only when you end a paragraph:

FOR YOUR INFORMATION
We are pleased to enclose details of exciting additions to our
range of financial services for the personal investor.
Our files show that you have an account with under the name ,
account number .
Please inform this office if any of these details have changed.

## Inserting Merge Fields

Now you can insert fieldnames from the database file so that Superbase will know which data goes where when you do the merge print run.

Click just before the word 'under'. Select the Edit Merge Field command, and when the field list requester appears click the **Bank_name** field. Superbase inserts &Bank_name& in the text:

Our files show that you have an account with &Bank_name&
under the name , account number .

Repeat the merge field insertion, positioning the **Name** field just before the comma, and the **Number** field just before the period:

Our files show that you have an account with &Bank_name&
under the name &Name&, account number &Number&.

If necessary edit the text until the paragraph is correct. Select Edit Reformat if you need to tidy up the paragraph.

Now save the text file. Choose Save As from the Text Editor's Project menu and name the file **Acc_memo.sbt** Click OK.

Return to the database window by selecting the Project Exit command.

## Running Mail Merge

Now that you have created a document into which Superbase can merge field data, you can run the merge operation from the Process menu. Before doing this, resize the Superbase window so that it occupies about half of the screen.

Select Process Mail Merge. Superbase displays a Filter requester. Click OK to pass through the requester. Since there is already a document in memory, Superbase assumes that it is to be used for the merge operation.

The next requester asks whether you wish to display or print the output. In this exercise we shall be content to preview the results, so click No to preview.

Superbase selects the first record from the **Accounts** file, merges the data from the **Bank_name, Name,** and **Number** fields into the **Acc_memo** document, and displays the results. The Pause button is switched on. Tap SPACEBAR to preview the next merged document. You can continue previewing for as many merged documents as you like. If you want to stop before the last record has been merged, click the Stop button or press CTRL+C .

Expand the window to its full size and click the current record button to redisplay the current record.

# Lesson 10: Globally Updating a File

Let's suppose there has been a general hike in interest rates of one per cent. You need to increase the value of the **Interest_rate** field by 1 in every record in the file that holds details of a Deposit account – type D accounts. You can do this in Superbase with the Process Update command.

Select Update Edit from the Process menu now. Superbase displays the Update filter requester. We need to build a filter command line that will select just those records that have 'D' or 'd' in the **Type** field. To do this we shall use the LIKE operator rather than the equals operator, because LIKE is case-insensitive – it will find both upper and lower case matches.

Move the pointer onto the fields list and click the **Type** field. Superbase copies it into the text box. Now click the LIKE button. In the Value box, type 'd' and press ENTER . The filter line should read:

    Type.Accounts LIKE "d"

If it does not, click Clear and start again. When the line is correct, click OK. Superbase displays the Update Definition requester.

Move the pointer onto the fields list and click **Interest_rate**. Superbase copies it into the text box and adds an equals sign. Now click **Interest_rate** again. The update command line reads:

    Interest_rate = Interest_rate

To complete the updating expression, click the + button and then type '1' in the Value box (omit quotes) and press ENTER . Your update command line should read:

    Interest_rate = Interest_rate + 1

What you have created is a command that instructs Superbase to add 1 to the value in each selected **Interest_rate** field. Superbase will read each record, carry out the command, and replace the record in the file with the updated value.

When your update command line reads correctly, click OK. Superbase quickly processes the whole file and returns to its normal state.

# Lesson 11: Querying the Database

The last lesson in this tutorial covers the Superbase Query command. This command is capable of creating sophisticated multi-file queries, but we shall only be using the example **Accounts** file. See Chapter 17 Multi-file Applications for more details of advanced features.

Query is used for producing reports with a more detailed structure than is possible with Table View or Process Print. You can control the layout of fields, add reporting functions such as totaling, and specify multi-level sorting, as well as creating filters like those used in earlier lessons. Let's see how this is done.

## Selecting Query

Select the Query command from the Process menu. Superbase displays the Query Definition requester. The requester has several controls. At the top are the controls for specifying a title for the query, which we shall explore next. Below these are four text boxes, with push buttons on their left. These buttons invoke the requesters for building up the different parts of the query: fields, report functions, filter, and sorting order. At the bottom left of the requester are the controls for specifying the destination for query output: screen, printer, disk file or Superbase file.

## Defining a Title

To define a title for the query, click in the text box at the top of the requester. Type 'Accounts List by Type'. Click the Page and Date check boxes to select them.

When the query is output, each page will include this title followed by one blank line, as well as the page number and system date.

## Selecting Fields for Output

Click the Fields button at the left of the requester. Superbase displays the Query Fields requester. You need to select some fields from the **Accounts** file, so move the pointer onto the fields list.

Click **Type, Reference, Name, Date_opened, Balance,** and **Interest** one after the other. Superbase copies the fields into the text box, separating them with commas:

> Type.Accounts, Reference.Accounts, Name.Accounts,
> Date_opened.Accounts, Balance.Accounts, Interest.Accounts

When the fields line is correct, click OK. Superbase copies the line into the Query Definition requester.

## Adding Reporting Functions

Invoke the Query Report requester by clicking the Report button. You are going to add some statistical functions to the query output.

First, click the Report button next to the fields list. This copies the REPORT keyword into the report line, indicating that the following functions are intended to operate on the report as a whole.

Now click Count to copy the COUNT keyword into the report line and request a count of the number of records output.

To request report totals, click Sum, then the **Balance** and **Interest** fields in the fields list. For averages, click Mean, then the same two fields, **Balance** and **Interest**.

The report line should read:

    REPORT COUNT SUM Balance.Accounts, Interest.Accounts
    MEAN Balance.Accounts, Interest.Accounts

You will not be able to see the entire line, but this does not matter.

Next, you need to request some reporting at subtotal level. We are going to sort and group the records by **Type**, so click Group to copy the GROUP keyword into the report line, then click **Type** in the fields list. This indicates that the following functions will apply to records when they are grouped on **Type**.

Now repeat the actions for REPORT , so your final report line should look like this:

    REPORT COUNT SUM Balance.Accounts, Interest.Accounts
    MEAN Balance.Accounts, Interest.Accounts GROUP
    Type.Accounts COUNT SUM Balance.Accounts,
    Interest.Accounts MEAN Balance.Accounts, Interest.Accounts

You are instructing Superbase to produce global and subtotal values for the **Balance** and **Interest** fields, and record counts for every group and the report as a whole.

When the report line is correct, click OK to copy it into the main Query Definition  requester.

## Defining a Filter

Queries may have the usual kinds of constraint applied through a filter command line. In this case, we shall simply exclude accounts which were opened before a certain date. Click the Filter button to invoke the Query Filter  requester.

Move the pointer onto the fields list and click **Date_opened**. Then click the
>= button, type '1/1/85' into the Value box and press ENTER . Your filter line
should read:

    Date_opened.Accounts  >=  "1/1/85"

Make sure the line is correct, then click OK to copy it into the Query
Definition requester.

## Defining a Sorting Order

The last query line to be defined is the Order line, which specifies how the
output is to be sorted. We shall ask for the records to be sorted in descending
order of the **Balance** field – from largest to smallest – and by **Type**. Click the
Order button to invoke the Query Order requester.

Click the **Type** field, the Ascending button, the **Balance** field, and the
Descending button. Superbase's Order line should read:

    Type.Accounts ASCENDING, Balance.Accounts DESCENDING

As you did before, click OK to copy the correct Order line to the main Query
Definition requester.

## Viewing Query Output

The default query output destination is the screen, and it is always advisable to
check your query definition visually before committing to what might be a long
print run.

Review the contents of the four text boxes. When you are satisfied that they
are what you intended, click OK. Superbase removes the Query Definition
requester and displays the results of the query on the screen.

When the output is broken by a subtotal level, as it is in this query, Superbase
prints the reporting results in a fixed format: subtotal, mean, record count .
The same format is used at the end of the query for the global report results.

## Saving the Query

Finally, save the query you have just created so that you can open it again for
reuse.

Select the Query Save As command from the Process menu. Type in a name
for the query, **Accounts**, and press ENTER . Superbase creates a disk file with an
SBQ extension, which will be listed every time you select the Process Query
Open command. The query file will open the **Accounts** file for you if it is not
already open.

See Chapter 14 Defining and Using Queries, for more details of this command.
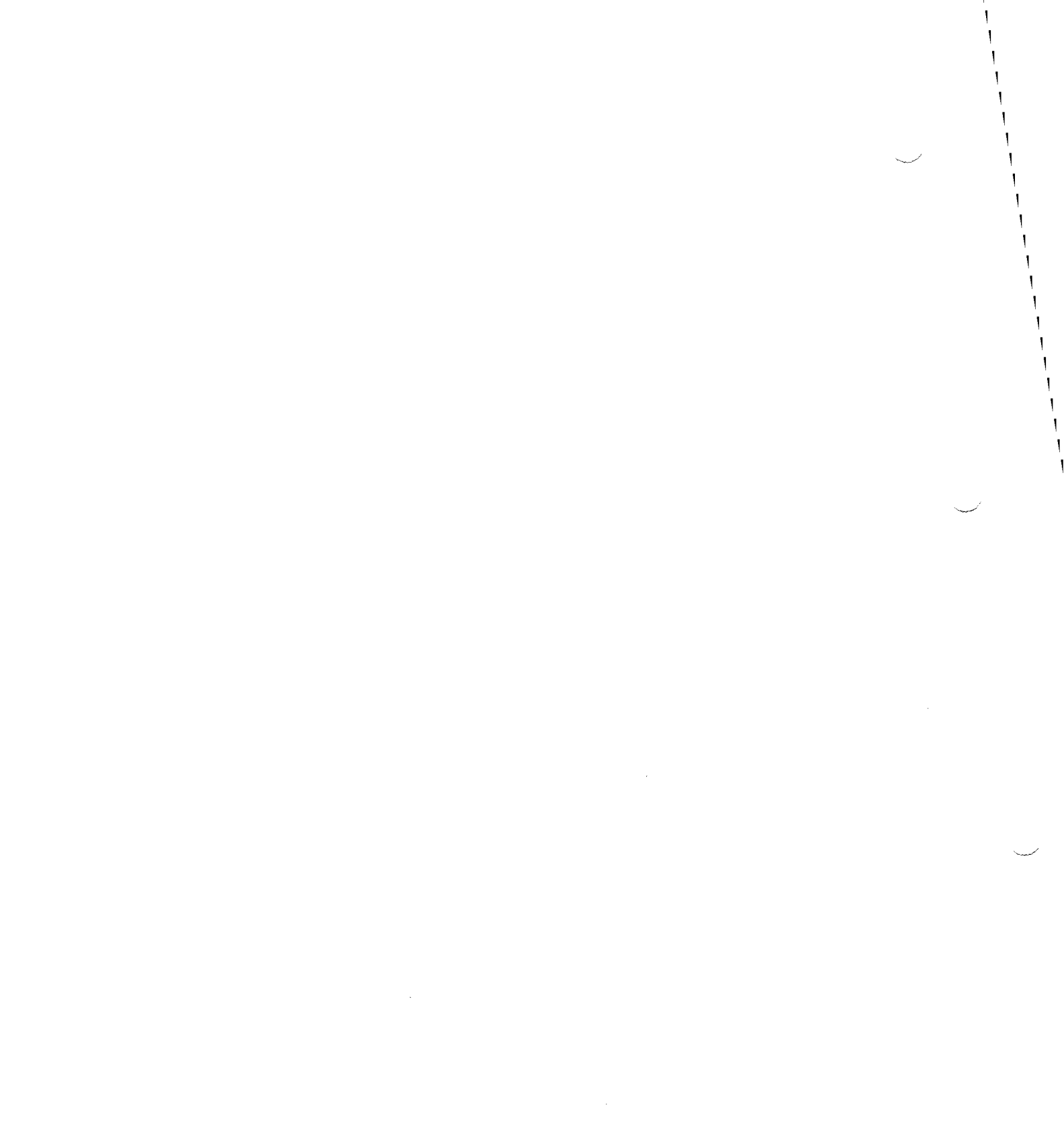
# 33 COMMAND REFERENCE SUMMARY

The first part of this chapter allows you to find out quickly what each Superbase menu command is for.

In the second part, there is a list of all the Superbase key controls grouped according to the following categories: menu selection, data entry, text and program editing, and record selection.

## Menu Commands

### Project Menu

The Project menu contains the main commands for managing the database: selecting the files and indexes for other menu commands to use.

**Project New File**
Create a new file. You enter the field names, formats, and attributes, plus any required formulas and input validation checks. The system then creates a file with an SBD extension to hold the file definition and a file with an SBF extension to hold the record data.

**Project New Empty Copy**
Create an empty copy of the current file. The system creates, under a new name, a file with an SBD extension holding a copy of the file definition of the current file and an empty file with an SBF extension to hold the record data.

**Project New Index**
Create a new index. Indexes can be used to display the records within a file in different orders, and for quick retrieval of records. Once you have created a new index, an index file will be created in the directory. This will have the same name as the file with an extension consisting of a period and a number.

**Project Open File**
Open an existing file. You can open more than one file at any one time, but only one can be the current file. The Query and Update functions can use multiple files. The number of files you can open depends on the amount of available memory.

**Project Open dBase File**
Open a file that was created by dBase. You can read such files and manipulate their data in various ways, but you cannot alter the contents of a dBase file from Superbase.

**Project Open Index**
Choose and open an existing index. The records in the current file are presented in the order of the open index.

**Project Open Form**
Open a form created in the Forms Designer. Any database files associated
with the form will be opened at the same time.

**Project Close File**
Close the current file. This command closes the current open file, reducing the
amount of memory needed by Superbase.

**Project Close Form**
Close the current form and switch to the most recently selected view mode.

**Project Close All**
Close all files and forms that are currently open.

**Project Modify File**
Change the definition of the current file. You can add or delete fields, change
field names, field lengths and field types, as well as any formulas or validation
checks.

**Project Modify Form**
Change the definition of the current form. You can add or delete fields,
change field names, lengths, types and their positions on the form, as well as
any formulas or validation checks.

**Project Save**
Save the current definition of the current file. If you have made any changes to
the file definition using File Modify and you wish to make them permanent,
you should do so using this command.

**Project Remove File**
Remove an open file from the disk. The SBF, SBD and all index files
associated with the selected file are deleted.

**Project Remove Index**
Remove an index from an open file. The relevant index file is deleted.

**Project Print Record**
Print the current record, using the current View format.

**Project Print File**
Print records from the current file. The function outputs the current fields to
the printer using the current View format. A Filter determines which records
are selected for printing. Ascending or Descending index order may be
selected.

**Project Print Status File**
Print a report of the status of the current file, including selected statistics. (See
also Utilities Status File.)

**Project Print Status System**
Print a short form report of the status of all currently open files. (See also Utilities Status System.)

**Project Print Directory**
Print a sorted list of the contents of your current directory. (See also Utilities Directory List.)

**Project Print Disk File**
Print the contents of a disk file. A requester is displayed allowing you to specify the directory and filename of the file.

**Project About**
Display details of the version of Superbase currently in use.

**Project Quit**
Exit from Superbase and return to Workbench.

## Edit Menu

The Edit menu contains all the commands for entering and editing data into a file.

**Edit Current**
Change the details of the current record within the current file. The record you wish to edit must be the current record. You can only edit those fields you have opened. If you are viewing the current record in Table format and you select this command, Superbase automatically switches you to Record or Page View and sets the Paging command on.

**Edit Cut**
Remove the current record (or field, when the insertion point is active in a field) and place it on the Clipboard.

**Edit Copy**
Copy the current record (or field) to the Clipboard.

**Edit Paste**
Paste the contents of the clipboard into the database file.

## Record Menu

**Record New**
Enter a new record into the current file. Superbase presents a blank record and you type the data into the fields. If you have opened a list of fields, then you can only enter information into those fields.

**Record Save**
Save the current record in the current file. This is used after you have entered record details by editing, or with the Record New or Record Duplicate commands.

### Record Duplicate

Make a copy in memory of the current record. This can then be edited and saved as a new record. The function saves time when you are entering the same or very similar information into more than one record.

### Record Next External

This command is used to manipulate the external files associated with a single record. This command enables you to display the next image within the current record, which could either be a picture or a text file.

### Record Previous External

This command is used to manipulate the external files associated with a single record. This command enables you to display the previous image within the current record, which could either be a picture or a text file.

## Process Menu

The Process menu commands cover all the main database processing requirements: searching, sorting, reporting, updating, deleting, importing, exporting, mail merge and label printing. All commands allow the use of a Filter to determine which records are to be processed.

### Process Query Edit

Run a Query, edit an existing Query which you have loaded from disk, or create a new one. Query allows you to interrogate the database and retrieve information from one or more files. You can include report formatting and analysis functions in the query. The results may be sorted in any order, and then output to the screen, the printer, a disk file, or a new database.

### Process Query Open

Choose and open a Query previously stored on the disk with Query Save. Superbase also opens the database files associated with the query.

### Process Query Save As

Save the settings for the current query. Superbase stores the title, settings for totalling, etc., and the Filter and Sorting instructions as they were last set up with Process Query Edit, in a SBQ extension file.

### Process Update Edit

Run an Update, edit an existing Update which has been loaded from disk, or create a new one. Use this command for automatic update of one or more fields within a set of records from one or more files. The Filter can refer to more than one file.

### Process Update Open

Load an Update from disk. Superbase also opens the database files associated with the update.

**Process Update Save As**
Save the current Update on disk. Superbase stores the update filter and fields setting on disk in a SBU extension file.

**Process Remove**
Remove a group of records from a file, according to the values and conditions set in the Remove Records Filter.

**Process Import**
Import data into a Superbase file from a range of different file types: two types of ASCII files, delimited and fixed length, dBase, Enable, Lotus, Logistix, Superplan and DIF files.

**Process Export**
Export data from a Superbase file. This command offers the same range of file types as Import, and converts the Superbase file to a file of the type specified.

**Process Split**
Using the data in the current file, generate two new database files with the same structure. You set a filter to determine how the records in the current file will be split between the two new files. The current file is left intact.

**Process Mail Merge**
Merge data from records in database file, with a form letter created in the Text Editor. The database must already be open; but if you haven't opened the text file which holds the form letter, Superbase will present a file requester where you can select a text file for your mail merge.

**Process Labels Edit**
Create new or edit existing label definition file.

**Process Labels Open**
Load an existing label definition file.

**Process Labels Save As**
Save current label definition. A requester is displayed allowing you to specify the filename to be used.

**Process Reorganize**
Reorganizes your database. This will be needed once you have been using your database for a while, and is used to tidy up the file space and make access time quicker.

## Set Menu

The Set menu contains a number of commands which directly affect the way data is processed and presented.

### Set Table View

Set the screen display to Table View. When you set this command the field names appear across the top of the screen, with the data for each field in the same column as the field name.

### Set Page View

Set the screen display to Page View. You can rearrange the line and column positions of the fields in Page View by dragging them around the screen.

### Set Record View

Set the screen display to Record View. When you set this command the field names will appear down the left hand side of the screen, with the data for each field on the same line as the field name.

### Set Form

Display a form if it is open.

### Set Field Selection

Select a list of fields. Once you have opened a file you can choose which fields to view, and so limit the amount of space each record takes up on the screen. This command enables you to choose those fields you want to see, for each record within the current file. If you have not selected a fields list, then all the fields for the current file will be displayed. The arrangement of fields last saved with File Save is recalled when you set this command.

### Set Paging

Switch Paging on or off. This allows you to choose whether you want to see the data in a continuous list (Paging off), or a page at a time (Paging on). The Edit menu data entry commands all switch Paging on automatically, as does Set Page View.

### Set Show Field Names

Switch field names on or off. This command whether field names are displayed or removed from the screen. It also applies to printed output.

### Set System Options

Lets you alter the system settings. These include: the speed at which records are scrolled onto the screen; the currency symbol; the format for thousands and decimal point separators; import and export field separators and external file handling controls; the start directory; and a number of other features.

### Set Printer Setup

Display requesters allowing you to select printer, page size, margins, typeface and other printing parameters from those available on your system.

**Set Number Format**
Choose the format for temporary numeric data such as the product of a
multiplication of two fields.

**Set Date Format**
Choose the format you want temporary dates or times to appear in, and set the
Superbase system date and time.

**Set Function keys Edit**
Create a new, or edit an existing, set of commands which may be invoked by
the function keys.

**Set Function keys Open**
Load an existing set of commands which may be invoked by the function keys.

**Set Function keys Save As**
Save the current set of commands which may be invoked by the function keys.
A requester is displayed allowing you to specify the filename to be used.

**Set Function keys List**
List the commands currently associated with the function keys.

## Utilities Menu

The Utilities menu includes a number of utilities to increase your control over
the database.

**Utilities Directory List**
Display a sorted list of the contents of the current directory. (See also Project
Print Directory.)

**Utilities Text Editor**
Select the Text Editor. This command opens the Text Editor window and
activates the Text Editor menus.

**Utilities Status File**
Display a report of the status of the current file, including selected statistics.
(See also Project Print Status File.)

**Utilities Status System**
Display a short form report of the status of all the open files. (See also Project
Print Status System.)

**Utilities Screen Dump**
Send an image of the current screen display to the printer.

**Utilities Communications**
Transfer a file to another computer. Allows you to send files to a remote
terminal using a modem, or to a computer nearby (local) using a cable
connection.

**Utilities Type**
Produces a listing of any text file in the directory. Note: This command can only be used with ASCII text files.

**Utilities Delete**
Delete any file on disk. Use this command as an alternative to the file deletion facility provided by your computer's operating system.

**Utilities Rename**
Rename any file on disk. Use this command as an alternative to the file rename facility provided by the computer's operating system.

**Utilities Copy**
Copy any file on disk. Use this command as an alternative to the file copying facility provided by the computer's operating system.

## DML Menu

The DML menu provides you with access to a number of DML commands from within Superbase.

**DML Command**
Display a requester allowing you to specify a DML command line of up to 256 characters, containing multiple DML commands if required.

**DML Run**
Run the current DML program (see DML Open).

**DML New**
Display the Program Editor screen, allowing you to type a new DML program and then to run it and/or store it in a specified program file.

**DML Open**
Load an existing DML program in order to run it.

**DML Close**
Close the current DML program.

**DML Edit**
Load an existing DML program in order to edit it.

**DML Save**
Save an edited DML program in a file of the same name as the file it was loaded from.

**DML Save As**
Save the current DML file. A requester is displayed allowing you to specify filename for the program file.

**DML Remove**
Delete an existing DML program. A requester is displayed allowing you to specify the filename of the program file to be deleted.

# Key Controls

## Superbase Menus

| Keys | Menu command |
|------|--------------|
| AMIGA+O | Project Open File |
| AMIGA+I | Project Open Index |
| AMIGA+Q | Project Quit |
| AMIGA+J | Project Open Form |
| AMIGA+M | Project Modify File |
| AMIGA+— | Project Close File |
| | |
| AMIGA+E | Edit Current |
| AMIGA+X | Edit Cut |
| AMIGA+C | Edit Copy |
| AMIGA+V | Edit Paste |
| | |
| AMIGA+N | Record New |
| AMIGA+S | Record Save |
| AMIGA+D | Record Duplicate |
| AMIGA+B | Record Batch |
| | |
| AMIGA+> | Edit Next External |
| AMIGA+< | Edit Previous External |
| | |
| AMIGA+? | Process Query Edit |
| AMIGA+W | Process Query Open |
| AMIGA+U | Process Update Edit |
| | |
| AMIGA+T | Set Table View |
| AMIGA+P | Set Page View |
| AMIGA+R | Set Record View |
| AMIGA+F | Set Form |
| AMIGA+= | Set Field Selection |
| AMIGA+K | Set Function Keys Edit |
| | |
| AMIGA+L | Utilities Dir List |
| AMIGA+Y | Utilities Status File |
| AMIGA+Z | Utilities Status System |
| AMIGA+H | Utilities Text Editor |
| | |
| AMIGA+A | DML Command |
| AMIGA+G | DML Run |
| AMIGA+/ | DML Edit |

## Browsing Controls

These keys are the equivalents of the browsing control buttons. They do not operate when an insertion point for data or text input is on screen.

| Key(s) | Browsing Control Button |
|---|---|
| RIGHT | Next record |
| LEFT | Previous record |
| UP/DOWN | Current record |
| SHIFT+DOWN | Fast forward |
| SHIFT+UP | Rewind |
| SPACEBAR | Pause/Release pause |
| CTRL+C | Stop |
| ? | Key Lookup |
| = | Filter |
| * | Read External File |
| CTRL+DOWN | Next page of transaction records |
| CTRL+UP | Previous page of transaction records |

## Data Entry

These controls are used for editing the data you enter in fields and the lines you enter in requesters; e.g., filters, query command lines, formulas, file and field names. Note that keys on the numeric keypad, such as HOME, END, INS and so on, operate as described below only when NUM LOCK is off (these keys otherwise give numeric values). You can switch NUM LOCK on and off in the Set System Options requester.

| Key(s) | Function |
|---|---|
| HOME or SHIFT+LEFT | Move insertion point to beginning of line |
| END or SHIFT+RIGHT | Move insertion point to end of line |
| CTRL+X | Clear field |
| CTRL+U | Undo field edits |
| ESCAPE | End data entry |
| RIGHT | Moves insertion point one character to the right. |
| LEFT | Moves insertion point one character to the left. |
| DEL | Deletes character to the right of insertion point. |
| BACKSPACE | Deletes character to the left of insertion point. |
| INS or CTRL+V | Insert or overtype |
| ENTER | Moves insertion point to the next field down or, on a form, to the next field in entry order. |
| CTRL+ENTER | Moves insertion point to the field above or, on a form, to the previous field in entry order. |

## Text Editor Controls

| Keys | Menu command |
|---|---|
| AMIGA+N | Project New |
| AMIGA+O | Project Open |

| | |
|---|---|
| AMIGA+S | Project Save |
| AMIGA+P | Project Print |
| AMIGA+M | Edit Mark Block |
| AMIGA+K | Edit Clear Block |
| AMIGA+X | Edit Cut |
| AMIGA+C | Edit Copy |
| AMIGA+V | Edit Paste |
| AMIGA+? | Search Search |
| AMIGA+= | Search Replace |
| AMIGA+> | Search Next |
| AMIGA+E | Options Plain |
| AMIGA+B | Options Bold |
| AMIGA+U | Options Underline |
| AMIGA+I | Options Italic |
| CTRL + B | Beginning of text |
| CTRL + W | Delete Word |
| CTRL + E | Delete to end of line |
| CTRL + N | New Line |
| CTRL + S | Split Line |
| CTRL + F | Reformat Paragraph |
| CTRL + G | End of text |
| CTRL + D | Delete Line |
| CTRL + X | Clear Line |
| CTRL + V | Insert On/Off |
| CTRL + A | Join Line |
| CTRL + U | Undo |
| SHIFT+UP | Scrolls text window up by one full screen |
| SHIFT+DOWN | Scrolls text window down by one full screen |
| TAB | Moves insertion point to the next multiple of eight characters to the right. |
| SHIFT+TAB | Moves insertion point to the next multiple of eight character to the left. |
| ENTER | In Typeover mode moves insertion point to start of next line; in Insert mode splits line at insertion point. |
| INS | Switches between Insert and Typeover modes. |
| BACKSPACE | Deletes the character to the left of insertion point. |
| DEL | Deletes the character to the right of insertion point. |
| CTRL+W | Deletes word to the right of insertion point. |

## Requester Response Keys

| Key | Meaning |
|---|---|
| RETURN | OK |
| ESC | Cancel |

## Selection Keys

| Key(s) | Meaning |
|--------|---------|
| + | Select next open file (to become current file) |
| – | Select next index to current file |
| CTRL+UP | Select next element in multiple response field |
| CTRL+DOWN | Select previous element in multiple response field |

# APPENDIX A: ERROR MESSAGES

- Error messages relating to the multi-user version of Superbase are dealt with in Chapter 19, Volume 2.
- Error message numbers are shown at the end of this appendix.

**A file must have at least one index**
If you need to remove and rebuild the only index to a file you must first build a temporary one on another field. To change from a unique to a duplicate index, you must remove and rebuild it.

**Access to file not allowed**
The file requires a password. Alternatively, you do not have the correct access privileges for the operation required.

**Ambiguous reference**
The OPEN FIELDS command includes a field which may not belong to the specified file.

**Array already dimensioned**
Program error. You have used the DIM statement twice for the same array.

**Array variable not dimensioned**
Program error. You have referred to an array variable without dimensioning it first using the DIM statement.

**Cannot AND – illegal values**
The 'AND' operator cannot handle values greater than 2,147,483,647 or less than -2,147,483,648.

**Cannot OR – illegal values**
The OR operator cannot handle values greater than 2,147,483,647 or less than -2,147,483,648.

**Cannot access directory**
System error or the directory doesn't exist. Check that the correct disk is inserted.

**Cannot access disk information**
System error. AmigaDOS cannot return directory information.

**Cannot divide by zero**
Divisor in an expression or equation is equal to zero.

**Cannot find this field**
Usually caused by mis-spelling a field name. Superbase treats any text item that is not a string variable or a reserved word as a field name. This message will be generated if the field referred to does not exist.

**Cannot find this file**
Usually caused by mis-spelling a file name in a command line. Superbase
treats a text item following a field name and a period (full stop) as file name.
This message will be generated if the file does not exist.

**Cannot open COM port**
Either your system does not provide a COM (serial) port, or the COM port
may be in use by another application.

**Cannot open the clipboard**
The clipboard is in use by another application.

**Cannot open the printer**
Physical system error. Check that the printer is correctly connected and
switched on.

**Cannot re-open**
Usually occurs when you click Cancel after modifying the file definition for a
dependent file (a file which is referenced by another open file). The solution is
to close the other file first.

**Cannot total non-numeric fields**
This error would occur if you attempted to generate a total using a text field;
for example, if you used the keyword SUM with a text field in the Query
Report command line.

**CASE required after SELECT CASE**
Program error. SELECT CASE must be accompanied by CASE.

**Closing parenthesis missing**
Error in formula, program or command line. There must be an equal number
of opening and closing parentheses.

**Command must be at the end of line**
Program error. Certain commands, such as SET, must not be followed by
other commands on the same line.

**Data too long for field**
Occurs when the data you are trying to store in a field is longer than the length
of the field as set in the file definition. Edit field length if necessary.

**Data types don't match**
Caused by combining different field types or literal data types in one
expression; for example, assigning a text string to a numeric field.

**DOS can't open file**
System error while trying to open a Superbase .SBF or index file.

**DOS can't open input file**
System error while trying to open a Superbase .SBD or other sequential file
for input.

**DOS can't open output file**
System error while trying to open a Superbase .SBD or other sequential file for output.

**DOS error deleting file**
Disk error while attempting to delete a file.

**DOS error reading data**
Disk read error while reading a Superbase .SBF, an index file, or other sequential files.

**DOS error reading file**
Disk read error while reading a Superbase .SBD or index file.

**DOS error writing data**
Disk error while writing a Superbase .SBF or index file.

**DOS error writing file**
Disk error while writing a Superbase .SBD or other sequential file.

**Duplicate entry for this index**
Index in use allows unique entries only. Can occur when saving a record or importing records leaving blank index fields.

**ELSE without END IF**
Program error. ELSE must be accompanied by END IF.

**End of external file**
The end of the file has been encountered unexpectedly. The file may be corrupt.

**End of file**
Information message.

**END SELECT missing from SELECT CASE**
Program error. SELECT CASE must be accompanied by END SELECT.

**Error during file transfer**
Hardware problem or disk error. If errors occurs in transmitting a block during file transfer, Superbase retries for a specified number of times. This error message is generated when the number of retries exceeds the maximum retry count.

**Error unlocking file**
Superbase system error. Quit and then reload Superbase.

**Error while renaming file**
Disk error. The most likely explanation is that the disk you are using has been corrupted. Try renaming the file to another disk, or saving it first in another directory.

**External file not compatible**
External picture files must be compatible with .GIF, .PCX or certain types of
.IFF files for Superbase to be able to display them.

**External file not found**
Occurs when the file named in an External File field is not found where it
should be. Check directory and/or path name and/or disk volume name.

**Field already exists**
Each field name in a File Definition must be unique.

**Field already has an index**
Occurs when you attempt to create an index on a field which already has an
index; for example, with the New Index option or with the CREATE INDEX
command.

**Field does not match validation**
If a field has a validation formula you must enter data within the formula's
limits. Errors can occur during Import or when editing records. Use Utilities
Status File to read Validation formulas.

**Field not defined**
Program error. You have referred to field which does not exist – if you type a
variable name incorrectly, Superbase will assume you are trying to access a
field and will issue this errror message.

**Field requires numeric data**
You cannot assign text data to a numeric field.

**Field requires text data**
You cannot assign numeric data to a text field.

**File already exists**
Superbase will not allow you to overwrite an existing file.

**File contains non-text characters**
Illegal control character detected in Superbase .SBD or other sequential file.

**File data in wrong format**
Illegal field and/or record separator detected during Import. Also occurs with
other sequential files when the format is incompatible with the function you
are trying to execute.

**File open in another directory**
A file cannot be opened if another file with the same name is open in another
directory.

**Floating point math error**
May occur during calculation of floating point values, for example due to
values being out of range.

**Formula too complex**
Evaluation capabilities have been exceeded.

**Function not allowed with number**
Mismatch of function and arguments. See Appendix B.

**Function not allowed with text**
Mismatch of function and arguments. See Appendix B.

**Function syntax – comma missing**
See Appendix B.

**Function syntax – invalid parameter**
See Appendix B.

**GOSUB or GOTO required in statement**
Program error. If a statement refers to a label on another line, the label must
be preceded by GOSUB or GOTO.

**Group not defined**
Program error. You have referred to a group which has not been defined in a
GROUP   statement.

**In use by another file**
You have attempted to close a file which is referenced by another open file.
Close the other file first.

**1.2 System software required**
To load forms, Superbase requires Amiga Workbench version 1.2 or later.

**Incomplete field or value missing**
Superbase has detected an error in a formula or a command line. Check that
expressions are complete and do not end with an operator.

**Input file not open**
Program error. You have used the INPUT command without first opening a
file for input with the OPEN FOR INPUT command.

**Insufficient memory**
Superbase cannot find sufficient memory for an operation such as viewing an
external file or sorting using Query Order.

**Invalid date**
The full date range is January 1, 0001 to December 31, 9999. You have
probably entered a date in the wrong day, month, year sequence giving a
month number greater than 12, or tried to enter illegal text or date separators.

**Invalid date format**
Program error, or SBD file invalid.

**Invalid field name**
Reserved words are not allowed: see Appendix D. Field names may not begin
with numbers, and should include only alphanumeric, space and underscore
characters.

**Invalid field type**
Program error, or SBD file invalid.

**Invalid form file**
Either the form file has been corrupted on disk, or you are attempting to load
a form that has not been created in the Form Designer.

**Invalid margins selected**
The values you have entered in the Margins requester exceed the limits for the
margin parameters. Alternatively, you may have entered a value for the left
margin which is greater than the right margin, or you may have entered a
negative value.

**Invalid numeric format**

Program error, or SBD file invalid.

**Invalid numeric parameter**
Illegal value for a function.

**Invalid parameter**
Program error. Check the syntax for the command or commands in the line
which caused the error.

**Invalid statement**
The DML EXECUTE command has encountered an unrecognized DML
keyword. This may originate from an incompatible version of Superbase.

**Invalid statement beginning**
Statements must begin with keywords, variables or labels. This message will be
issued if you start a statement with anything else, e.g. an arithmetic operator.

**Invalid statement ending**
Usually means that you have entered an operator at the end of the statement
but have not followed it with a value of some kind, e.g. ending a statement with
'+' would cause this error message. After executing a statement, Superbase
expects to find either a colon (followed by another command) or an end of
line. It will generate this message if it finds an extraneous keyword, variable, or
label.

**Invalid subscript for array**
Program error. Array subscript exceeds the number of elements specified for
the array in a DIM statement.

**Invalid text parameter**
Illegal value for a function.

**Invalid time**
The time range is from 0:00 to 23:59 (or 11:59 pm). You have entered a time which falls outside this range, e.g. 25:06.

**Key not found**
Appears for your information whenever a key lookup doesn't find an exact key match, even if you used a partial key entry to retrieve the record.

**Logical field must contain Y, N, T or F**
The contents of a logical field are restricted as indicated.

**Misplaced command**
Operand missing in command line, or a keyword has been used in a command where it is inappropriate.

**Name too long**
Field or variable exceeds limit of 15 characters.

**Nesting depth exceeded**
Superbase supports up to 50 nesting levels with GOSUB, FOR NEXT, and WHILE WEND statments. You may have exceeded this limit. But a more likely explanation is that you have set up a Report Group in the wrong place in a Report program, e.g in a Report subroutine.

**NEXT without FOR**
Program error. If the program contains a number of nested FOR NEXT loops, check that each FOR has a corresponding NEXT.

**No carrier detected (DCD)**
DCD must be high during file transfer.

**No external file field defined**
You have clicked on the External File (camera) button when the current file has no External File fields in its definition.

**No file name given**
Program error. You have used a statement which requires a file name.

**No file selected**
Program error. You have used a statement which requires a file to be open.

**No more images for this record**
Next External or Previous External cannot work when you are already looking at the last or first image respectively for the current record.

**No record selected**
Program error. You have referred to a record without having opened the file it belongs to.

**Not an external field**
Program error. The SHOW command must refer to an external field.

**Not an index**
Your attempt to open or remove an index failed because the index file does not exist.

**Not an open file**
File named in command line is not an open database file.

**Not enough memory to run application**
Too many Workbench applications are open to run Superbase. Close down other applications.

**Number too large**
Superbase handles numbers up to 1.797693134862e308.

**Opening parenthesis missing**
Error in formula, prograam or command line. There must be equal number of opening and closing parentheses.

**Operator not allowed with text**
Only certain operators may be used with text items; e.g. you cannot multiply text items.

**Out of data**
Program error. This message is issued when there is no more data for a READ statement. It usually means that the data items in the programs DATA statements have already been read.

**Print command failed**
Failure occurred during a printing operation.

**Problem positioning in data file**
Superbase system error. Occurs when Superbase tries to read a record which does not exist. The cause may be that you have swapped disks without first closing files or changing the directory. You should always close any open files before changing disks. The first step you should take is to make a backup copy of the file, then reorganize it.

**Problem reading external file**
Error within external file.

**Remove index before changing field type**
Superbase does not allow you to change the type of an indexed field. Use the Remove Index command on the Project menu to remove the index from the field.

**Remove index before deleting field**
You cannot remove an indexed field from the File Definition until you have made it a non-indexed field by removing the index.

**Required in statement**
Program error. A command which is required by the syntax of the statement has been omitted, e.g. this message would be issued if you left out the equal sign in a statement which assigned a value to a variable.

**RESUME without error**
The RESUME command cannot be issued if an error has not occurred.

**RETURN without GOSUB**
Program error. Subroutines must be called using the GOSUB or ON GOSUB statements. This error message indicates that the program control has passed to a subroutine by other some route. Possibly you have used GOTO instead of GOSUB or, more commonly, program control has simply 'dropped through' to the subroutine because you have failed to direct it elsewhere.

**Search string not found**
The text string you are searching for does not exist in the current file.

**Statement does not give TRUE or FALSE**
You have defined a validation formula which cannot be evaluated. This error would occur, for example, if the formula did not contain an operator.

**Statement has an unrelated section**
An expression in a command or program line is not joined to another by AND or OR.

**String too long**
Superbase application program error. Maximum length of a text string stored in a field or a variable is 4000 characters. Some strings, such as command lines, are limited to 255 characters.

**Subscript not numeric**
You have entered a string value as an array subscript.

**Error deleting key: 100/101/102/103**
Superbase system error in index. Backup the file, then reorganize it.

**Error reading block: 200/201**
Superbase system error in data file. Backup the file, then reorganize it.

**THEN or GOTO required in statement**
Program error. The keyword IF must be followed by THEN or GOTO.

**This field must have some data**
This field is a required field. If a file contains required fields, you must enter data in these fields when you are creating a new record.

**This WHERE statement must be a single file**
Some Superbase functions, such as Labels, do not support multi-file application. This message occurs when you use the WHERE statement in a single file function to refer to multiple files.

**Undefined error**
Internal Superbase system error. Please notify software supplier.

**Undefined program label**
Program error. The label following a GOSUB, ON GOSUB, GOTO, or ON GOTO statement does not exist elsewhere in the program.

**Variable not defined**
Program error. Occurs when you use a variable in a statement without having assigned it a value beforehand.

**Variable required in statement**
Certain commands such as INPUT and GET must be followed by a variable.

**WEND without WHILE**
Program error. WEND must be accompanied by WHILE.

**Write failed - Disk is full**
Your data file may be unstable. Use Process Reorganize to transfer data to another volume and/or file.

# Error Message Numbers

| Number | Message Text |
|---|---|
| 1 | Not an index |
| 2 | No more images for this record |
| 3 | Print command failed |
| 4 | Cannot open the printer |
| 5 | Not an open file |
| 6 | Access to file not allowed |
| 7 | A file must have at least one index |
| 8 | Key not found: Next in sequence selected |
| 9 | No external file field defined |
| 10 | End of file |
| 11 | (No message) |
| 12 | Cannot access directory |
| 13 | Cannot access disk information |
| 14 | Name too long |
| 15 | Cannot find this file |
| 16 | Cannot find this field |
| 17 | Invalid statement |
| 18 | Out of data |
| 19 | Invalid date |
| 20 | Data types don't match |
| 21 | Operator not allowed with text |
| 22 | Function not allowed with text |
| 23 | Function not allowed with number |

| | |
|---|---|
| 24 | Cannot divide by zero |
| 25 | Cannot 'AND' -- illegal values |
| 26 | Cannot 'OR' -- illegal values |
| 27 | Invalid numeric parameter |
| 28 | Invalid text parameter |
| 29 | Closing parenthesis missing |
| 30 | Opening parenthesis missing |
| 31 | Function syntax -- comma missing |
| 32 | Function syntax -- invalid parameter |
| 33 | String too long |
| 34 | Misplaced command |
| 35 | Formula too complex |
| 36 | Variable not defined |
| 37 | Field not defined |
| 38 | Number too large |
| 39 | Invalid numeric format |
| 40 | No file name given |
| 41 | Invalid parameter |
| 42 | Field already exists |
| 43 | File already exists |
| 44 | No file selected |
| 45 | Invalid field type |
| 46 | Invalid field name |
| 47 | Data too long for field |
| 48 | Field requires numeric data |
| 49 | Field requires text data |
| 50 | This field must have some data |
| 51 | Field does not match validation |
| 52 | DOS cannot open file: |
| 53 | DOS error deleting file: |
| 54 | DOS error writing file: |
| 55 | RETURN without GOSUB |
| 56 | NEXT without FOR |
| 57 | Duplicate entry for this index |
| 58 | No record selected |
| 59 | SUPERBASE error deleting key: 100 |
| 60 | SUPERBASE error deleting key: 101 |
| 61 | SUPERBASE error deleting key: 102 |
| 62 | SUPERBASE error deleting key: 103 |
| 63 | Memory allocation error |
| 64 | DOS error reading file: |
| 65 | DOS error writing file: |
| 66 | File contains non-text characters |
| 67 | File data in wrong format |
| 68 | DOS cannot open input file: |

| | |
|---|---|
| 69 | DOS cannot open output file: |
| 70 | Undefined program label |
| 71 | SUPERBASE error reading block: 200 |
| 72 | SUPERBASE error reading block: 201 |
| 73 | Invalid date format |
| 74 | Invalid numeric format |
| 75 | DOS error reading data: |
| 76 | DOS error writing data: |
| 77 | Problem positioning in data file |
| 78 | End of external file |
| 79 | External file not found |
| 80 | External file not compatible |
| 81 | Problem reading external file |
| 82 | Remove index before deleting field |
| 83 | Incomplete -- field or value missing |
| 84 | Statement has unrelated section |
| 85 | Statement does not give TRUE or FALSE |
| 86 | Remove index before changing field type |
| 87 | Cannot total Non-Numeric fields |
| 88 | Insufficient memory |
| 89 | Dongle missing from games port 2 |
| 90 | Insufficient stack space |
| 91 | Error while renaming file |
| 92 | This WHERE statement must be single file |
| 93 | Command must be at end of line |
| 94 | Invalid time |
| 95 | Input file not open |
| 96 | Not an external field |
| 97 | RESUME without error |
| 98 | Problem with narrator device |
| 99 | Problem with console device |
| 100 | Field already has an index |
| 101 | Array variable not dimensioned |
| 102 | Invalid subscript for array |
| 103 | Subscript not numeric |
| 104 | Array already dimensioned |
| 105 | WEND without WHILE |
| 106 | Nesting depth exceeded |
| 107 | Problem loading overlay |
| 108 | Invalid statement beginning |
| 109 | Invalid statement ending |
| 110 | Required in statement |
| 111 | Variable required in statement |
| 112 | THEN or GOTO required in statement |
| 113 | GOSUB or GOTO required in statement |

| | |
|---|---|
| 114 | Invalid form file |
| 115 | Group not defined |
| 116 | Problem opening font library |
| 117 | File open in another directory |
| 118 | Problem opening window |
| 119 | 1.2 System software required |
| 120 | Ambiguous reference |
| 121 | ELSE without END IF |
| 122 | In use by another file |
| 123 | Cannot re-open |
| 124 | Cannot open COM port |
| 125 | Error during file transfer |
| 126 | No carrier detected (DCD) |
| 127 | File is busy |
| 128 | Error unlocking file |
| 129 | Record in use |
| 130 | Record being modified |
| 131 | File is open for exclusive access |
| 132 | File is open for shared access |
| 133 | Too many users for this version |
| 134 | Network or file sharing inactive |
| 135 | File has changed: Select record by key |
| 136 | Exclusive lock required |
| 137 | File in use by: |
| 138 | Access denied: Sharing violation |
| 139 | File is open for exclusive write |
| 140 | File is open for shared write |
| 141 | File has been modified |
| 142 | File is open for read only |
| 143 | ERR |
| 144 | ERR |
| 145 | ERR |
| 146 | ERR |
| 147 | ERR |
| 148 | Search string not found |
| 149 | ERR |
| 150 | Invalid margins selected |
| 151 | CASE required after SELECT CASE |
| 152 | END SELECT missing from SELECT CASE |
| 153 | Write failed - Disk is full |
| 154 | ERR |
| 155 | Logical field must contain Y,N,T or F |
| 156 | ERR |
| 157 | ERR |
| 158 | Cannot open the Clipboard |

| | |
|---|---|
| 159 | Floating point math error |
| 160 | ERR |
| 161 | Precision error in math function |
| 162 | Form requires new version of software |
| 163 | Reorganize failed to complete |
| 164 | Insufficient disk space |
| 165 | ERR |
| 166 | ERR |
| 167 | ERR |
| 168 | ERR |
| 169 | Invalid Superbase library |
| 170 | Network control file not present |
| 171 | Invalid network control file directory |
| 172 | Cannot access network control file |
| 173 | Invalid lock file |
| 174 | Cannot lock database file |
| 175 | Cannot lock file |
| 176 | Cannot lock record |
| 177 | Record is not locked |

# APPENDIX B: FUNCTIONS

The terms in italics indicate the type of argument each function takes:

$X$ and $N$ must be numeric values. The value can be supplied by a numeric field, a numeric constant (i.e. a number), or the result of a calculation or another function. Examples are:

    INT(Total1/Total2)
    INT(235/52)
    INT(SQR(numfield))

$X\$$ must be a text string. You can enter the name of a text field, a text constant (ie. a string of characters enclosed in quotation marks), a concatenated string, or the result of another function. Examples are:

    RIGHT$("MrSmith",4)
    RIGHT$(Name.Address,4)
    LTRIM$(Code + "/" + LEFT$(Lastname,3))

The other terms - *file name, date, time*, etc - should be self explanatory. File names must be enclosed in quotation marks. The values for *date* and *time* can be supplied by date and time fields, the results of calculations involving date and time fields, or system variables (TODAY and NOW).

Some functions test whether something is true or false. EXISTS, for example, checks whether a file or an index value exists. In such cases, the function returns -1 for a true result, 0 for a false result.

ABS(*X*)
Returns the positive value of *X*.

ASC(*X$*)
Returns the ASCII value of a single character.

ATN(*X*)
Returns the angle measured in radians, whose tangent is *X*.

CHR$( *X*)
Returns the text character associated with the ASCII code value in *X*.

COL( *0*)
Returns the current screen column.

COS(*X*)
Returns the cosine value of *X*, where *X* is an angle in radians.

DATE$( *date [,date format]*)
Returns a text string from a Superbase date field or a julian date number, using optional format.

DAY$( *date* )
Returns the day of the week as a text string.

DAY( *date* )
Returns the day of the month as a numeric value from a date field or a date string.

DAYS( *date* )
Returns the julian date number of a date field or date string.

DISKSPACE( *disk name* )
Returns the amount of free disk space.

EOF( *filename* )
Returns the value of -1 when the end of a file is reached.

ERR$( *N* )
Returns the error message for error number *N*.

EXISTS( *X$* [,*indexfield*])
*X$* may either be a file name or an index value. Returns -1 (TRUE) if the file exists on disk or the value exists in a the specified index field, otherwise returns 0 (FALSE).

EXP( *X* )
Returns the mathematical constant 'e' (*2.71827183*) raised to the power of *X*.

FCASE$( *X$* )
Capitalizes the first letter of *X$*.

FIX( *X,N* )
Sets the number of decimal places to which the numeric expression X (which may be a field) is stored.

FN alpha(*strexpr*)
Returns a string containing only alphabetic characters (a-z, A-Z).

FN ansi(*strexpr*)
Returns a string in which all IBM characters are converted to the ANSI character set.

FN dec (*strexpr*)
Returns the decimal equivalent of a string into a number.

FN ext (*strexpr*)
Returns a string containing the extension of a filename.

FN fact(*X*)
Returns the factorial of a number.

FN fv(*rate,nper,pmt*[,*pv*[,*type*]])
Returns the future value of an investment.

FN hex *(nexpr)*
Returns a string containing the hexadecimal equivalent of a decimal.

FN ibm*(strexpr)*
Returns a string in which all ANSI characters are converted to the IBM
character set.

FN name *(strexpr)*
Returns a string containing a filename together with its extension.

FN nper*(rate,pmt,pv[,fv[,type]])*
Returns the number of payments on an investment.

FN numeric*(strexpr)*
Returns a string containing only numeric characters (0-9).

FN path *(strexpr)*
Returns a string containing all pathname characters including final backslash.

FN pmt*(rate,nper,pv[,fv[,type]])*
Returns the value of a periodic payment, such a loan repayment.

FN pv*(rate,nper,pmt[,fv[,type]])*
Returns the present value of an investment.

FN rate*(nper,pmt,pv[,fv[,type[,guess]]])*
Returns the interest rate per period for an investment.

FN root *(strexpr)*
Returns a string containing the root of a filename.

FN sln( *cost, salvage, life)*
Returns straight line depreciation for an asset for a single period.

FN sys*(nexpr)*
Displays one of 17 items of system information. *nexpr* (in the range 0-16)
determines which. See DML Commands, in Volume 2, for details.

FORMAT$ *(strexpr, nexpr1 [,nexpr2])*
Forces query and report output to wordwrap within a column.

FOUND( *file name)*
Returns a numeric value which depends on the result of the last key lookup
search. If the search was successful, it returns -1 (TRUE), otherwise it returns
0 (FALSE).

FREE( *N)*
Returns the amount of free memory. *N* is a dummy variable and can take any
value.

HRS( *time )*
Returns the number of hours from a time field.

INSTR( *N,X$,Y$* )
Returns *N* as the starting character position of substring *Y$* within the string
*X$*, or 0 if the string is not found.

INT(*X*)
Returns the integer value of a number.

LCASE$( *X$*)
Returns the lower case equivalent of a text string or field.

LEFT$( *X$,N*)
Returns the leftmost *N* characters of a text string or field.

LEN( *X$*)
Returns the length of a text string or field.

LOG( *X*)
Returns the natural log of a numeric field or number.

LOOKUP( *field1,field2*)
Validates the data in field1 by checking to see if it exists in field2. Field2 must
be a field in another file.

LTRIM$( *X$*)
Trims leading spaces from *X$*.

MID$( *X$,M,N* )
Returns a substring of length *N* beginning at character position *M* from within
a text string or field.

MINS( *time* )
Returns the number of minutes from a time field.

MONTH$( *date*)
Returns the month of the year as a text string.

MONTH( *date*)
Returns a numeric value for the month of the year.

PAD$( *X$/textfield,[N]*)
Returns text string or field *N* characters long. If a field and *N* is omitted, takes
the length set in the file definition.

PCOL( *0*)
Returns the current printer column.

PROW( *0*)
Returns the current printer row.

RECCOUNT( *file name*)
Returns the number of records in file.

REPLICATE( *X$,N* )
Replicates *X$* N times.

RIGHT$( *X$,N* )
Returns the rightmost *N* character of a text string or field.

RND( *X* )
Returns a randomly generated number between 0 and 1. If *X* is 0 the number is the same as the previous one. If X is less than zero, the function is reseeded. If *X* is greater than zero, a new number is generated.

SECS( *time* )
Returns the number of seconds left over after subtracting the number of hours and minutes.

SER( *file name* )
Returns the serial number of the specified file.

SGN( *X* )
Returns a value depending on the value of *X*: -1 for negative, 0 or zero, 1 for positive.

SIN( *X* )
Returns the sine of *X* where *X* is a value in radians.

SPACE$( *N* )
Returns a text string with *N* spaces.

SQR( *X* )
Returns the square root of *X*.

STR$( *X [N1 [,N2]] / [X$]* )
Returns the text string equivalent of a number or numeric field represented by *X*. The arguments *N1* and *N2* can be used to specify the numeric format of the result: the first gives the number of integers before the decimal point, the second gives the number of digits after the decimal point. *X$* determines the numeric format.

TAN( *X* )
Returns the tangent of *X* where *X* is a value in radians.

THOUSECS( *time* )
Returns the number of thousandths of a second left over after subtracting seconds, minutes and hours.

TIME$( *X,time format* )
Returns a time string from *X* using optional time format.

TIMEVAL( *time* )
Return the value of a time string in thousandths of a second.

TRIM$( *X$* )
Trims trailing spaces from *X$*.

UCASE$( *X$* )
Converts a text string to upper case.

VAL( *X$* )
Returns the numeric equivalent of a text string or field.

YEAR( *date* )
Returns a numeric value for the year.

# APPENDIX C: RESERVED WORDS

| | | |
|---|---|---|
| ABS | DDE$ | FORM |
| ADD | DDEEXEC | FORMAT$ |
| AFTER | DDEINIT | FOUND |
| ALL | DDEPOKE | FREE |
| AND | DDEREQ | FROM |
| APPEND | DDETERM | GET |
| AS | DEBUG | GOSUB |
| ASC | DELETE | GOTO |
| ASCENDING | DESCENDING | GROUP |
| ASK | DIM | HEADING |
| ATN | DIRECTORY | HOME |
| ATTR | DISK | HRS |
| BEFORE | DISKSPACE | IF |
| BELL | DISPLAY | IMPORT |
| BF | DOWN | INDEX |
| BG | DUPLICATE | INPUT |
| BLANK | EDIT | INSTR |
| BREAK | EJECT | INT |
| BUFFERS | ELSE | IS |
| CALL | END | IT |
| CASE | ENTER | KEY |
| CHAIN | EOF | LABELS |
| CHR$ | ERASE | LAST |
| CLEAR | ERR$ | LCASE$ |
| CLOSE | ERRNO | LEFT$ |
| CLS | ERROR | LEN |
| COL | EXECUTE | LET |
| COMMS | EXISTS | LIKE |
| CONTAINS | EXP | LINE |
| COPY | EXPORT | LIST |
| COS | FCASE$ | LOAD |
| COUNT | FG | LOCATE |
| CREATE | FIELD | LOCK |
| CURRENT | FIELDS | LOG |
| DATA | FILE | LOOKUP |
| DATE$ | FIRST | LTRIM$ |
| DATEBASE | FIX | MACRO |
| DAY | FN | MAKE |
| DAY$ | FOOTING | MAX |
| DAYS | FOR | MEAN |

| | | |
|---|---|---|
| MEMORY | PRINT | SIN |
| MENU | PRINTER | SPACE$ |
| MERGE | PROTECT | SQR |
| MID$ | PROW | STATUS |
| MIN | QUERY | STEP |
| MINS | QUIT | STORE |
| MOD | READ | STR$ |
| MODIFY | RECCOUNT | SUM |
| MONTH | RECORD | SUMMARIZE |
| MONTH$ | REM | TABLE |
| MOUSE | REMOVE | TAN |
| NEW | RENAME | TEXT |
| NEWLINE | REORGANIZE | THEN |
| NEXT | REPLICATE | THOUSECS |
| NOT | REPORT | TIME$ |
| NOW | REQUEST | TIMEVAL |
| NUMBASE | RESTORE | TO |
| OFF | RESUME | TODAY |
| ON | RETURN | TRIM$ |
| OPEN | RIGHT$ | UCASE$ |
| OR | RND | UL |
| ORDER | ROW | UNIQUE |
| OUTPUT | RUN | UPDATE |
| PAD$ | SAVE | USERNAME |
| PAGE | SAY | USING |
| PAGING | SCRDUMP | VAL |
| PANEL | SD | VAR |
| PASSWORD | SECS | VIEW |
| PCOL | SELECT | WAIT |
| PG | SER | WEND |
| PI | SET | WHERE |
| POSITION | SGN | WHILE |
| POST | SHARE | YEAR |
| PREVIOUS | SHOW | |

# APPENDIX D: ASCII VALUES

| ASCII | Decimal | ASCII | Decimal |
|-------|---------|-------|---------|
| NUL | 0 | Space | 32 |
| SOH | 1 | ! | 33 |
| STX | 2 | " | 34 |
| ETX | 3 | # | 35 |
| EOT | 4 | $ | 36 |
| ENQ | 5 | % | 37 |
| ACK | 6 | & | 38 |
| BEL | 7 | ' | 39 |
| BS | 8 | ( | 40 |
| HT | 9 | ) | 41 |
| LF | 10 | * | 42 |
| VT | 11 | + | 43 |
| FF | 12 | , | 44 |
| CR | 13 | - | 45 |
| SO | 14 | . | 46 |
| SI | 15 | / | 47 |
| DLE | 16 | 0 | 48 |
| DC1 | 17 | 1 | 49 |
| DC2 | 18 | 2 | 50 |
| DC3 | 19 | 3 | 51 |
| DC4 | 20 | 4 | 52 |
| NAK | 21 | 5 | 53 |
| SYM | 22 | 6 | 54 |
| ETB | 23 | 7 | 55 |
| CAN | 24 | 8 | 56 |
| EM | 25 | 9 | 57 |
| SUB | 26 | : | 58 |
| ESC | 27 | ; | 59 |
| PS | 28 | < | 60 |
| GS | 29 | = | 61 |
| RS | 30 | > | 62 |
| US | 31 | ? | 63 |

| ASCII | Decimal | ASCII | Decimal |
|-------|---------|-------|---------|
| @ | 64 | ' | 96 |
| A | 65 | a | 97 |
| B | 66 | b | 98 |
| C | 67 | c | 99 |
| D | 68 | d | 100 |
| E | 69 | e | 101 |
| F | 70 | f | 102 |
| G | 71 | g | 103 |
| H | 72 | h | 104 |
| I | 73 | i | 105 |
| J | 74 | j | 106 |
| K | 75 | k | 107 |
| L | 76 | l | 108 |
| M | 77 | m | 109 |
| N | 78 | n | 110 |
| O | 79 | o | 111 |
| P | 80 | p | 112 |
| Q | 81 | q | 113 |
| R | 82 | r | 114 |
| S | 83 | s | 115 |
| T | 84 | t | 116 |
| U | 85 | u | 117 |
| V | 86 | v | 118 |
| W | 87 | w | 119 |
| X | 88 | x | 120 |
| Y | 89 | y | 121 |
| Z | 90 | z | 122 |
| [ | 91 | { | 123 |
| \ | 92 | | | 124 |
| ] | 93 | } | 125 |
| ^ | 94 | ~ | 126 |
| - | 95 | DEL | 127 |

# APPENDIX E: SUPERBASE 4 FILE TYPES

The Superbase system consists of program files and some ancillary work files. Database files are of three different kinds, and there are other types of application files as well.

## System Files

SBPRO4.EXE        Superbase executable program file

SBFD4.EXE        Superbase Forms Designer executable program file

## Database Files

filename.SBD        Definition file. A text file containing formatting information for field data.

filename.SBF        Data file. The file in which record data is stored in a series of linked blocks.

filename.1 through filename.999
        Index files for each database file. Indexes are automatically maintained.

## Other Application Files

filename.SBV        Form file. Created with Forms Designer. May invoke a report program.

filename.SBP        Program file. Contains Superbase DML program code.

filename.SBT        Text Editor file. Contains margin format control codes.

filename.SBQ        Query file. Contains instructions for field selection, reporting features, file links, filter conditions, and sorting order.

filename.SBU        Update file. Contains instructions for global file updating: file links, filter conditions, and update actions to be performed.

filename.SBB        Labels file. Contains details of the labels format specified with the Process Labels command.

filename.SBK        Function key file. Stores up to 40 function key assignments.

filename.DBD        dBase definition file. Contains details of the file structure of a dBase file and of any indexes associated with the file.

# INDEX

## Index